

IBM® WebSphere Commerce



Guida per lo sviluppatore del negozio

Versione 54

IBM® WebSphere Commerce



Guida per lo sviluppatore del negozio

Versione 54

Nota:

Prima di utilizzare queste informazioni e il prodotto supportato, consultare il paragrafo Informazioni particolari.

Seconda edizione (maggio 2002)

Questa edizione si applica ai seguenti prodotti:

- IBM WebSphere Commerce Business Edition per Windows NT e Windows 2000, Versione 5.4
- IBM WebSphere Commerce Business Edition per AIX, Versione 5.4
- IBM WebSphere Commerce Business Edition per Solaris, Versione 5.4
- IBM WebSphere Commerce Studio, Business Developer Edition per Windows NT e Windows 2000, Versione 5.4
- IBM WebSphere Commerce Professional Edition per Windows NT e Windows 2000, Versione 5.4
- IBM WebSphere Commerce Professional Edition per AIX, Versione 5.4
- IBM WebSphere Commerce Professional Edition per Solaris, Versione 5.4
- IBM WebSphere Commerce Studio, Professional Developer Edition per Windows NT e Windows 2000, Versione 5.4

e a tutti i successivi rilasci e modifiche di questi prodotti, se non diversamente specificato in nuove edizioni. Accertarsi di utilizzare l'edizione corretta per il livello del prodotto.

Ordinare le pubblicazioni tramite il rappresentante IBM o presso la filiale IBM di zona. Le pubblicazioni non sono disponibili all'indirizzo di seguito riportato.

Come ultima pagina del manuale è stato predisposto un foglio riservato ai commenti del lettore. Se il foglio è stato rimosso, i commenti possono essere inviati al seguente indirizzo:

SELFIN S.p.A.
Translation Assurance
Via F. Giordani, 7
80122 Napoli
ITALY

Tutti i commenti e i suggerimenti inviati potranno essere utilizzati liberamente dall'IBM o dalla Selfin e diventeranno esclusiva delle stesse.

© Copyright International Business Machines Corporation 2002. Tutti i diritti riservati.

Indice

Prima di iniziare.	vii
Convenzioni utilizzate in questo manuale	vii
Dove reperire ulteriori informazioni	viii

Parte 1. Panoramica 1

Capitolo 1. Panoramica sull'architettura del negozio	3
Che cos'è un negozio in linea?	3
Composizione di un negozio	3
Architettura del negozio	4

Capitolo 2. Sviluppo del negozio 7

Opzioni per lo sviluppo del negozio	7
Creazione di un negozio basato su un esempio	7
Creazione di un negozio sviluppando nuove risorse del negozio	8
Creazione di un negozio utilizzando una combinazione del negozio di esempio e delle nuove risorse del negozio	8
L'archivio negozio	8
Archivi negozio di esempio	9
Quando utilizzare un archivio negozio.	9
Strumenti di sviluppo del negozio.	10
Strumenti per lo sviluppo della facciata del negozio.	10
Strumenti per lo sviluppo dei dati del negozio.	11
Strumenti per lo sviluppo dell'ufficio posteriore	12
Il ruolo dello sviluppatore del negozio	12

Parte 2. Sviluppo della facciata del negozio 15

Capitolo 3. Sviluppo della facciata del negozio 17

Architettura della facciata del negozio	17
Comandi e viste predefiniti	18
Creazione di pagine del negozio	18
Sviluppo di un elenco di pagine del negozio	18
Sviluppo di un elenco di URL di comandi e viste	21
Associazione di nomi file JSP e viste	22

Parte 3. Panoramica sui dati del negozio 25

Capitolo 4. Dati del negozio. 27

Cosa sono i dati del negozio.	27
Modello di informazioni dati del negozio	27
Risorse dati del negozio	28
Architettura dati del negozio	29
Architettura dati del negozio e negozi di esempio	31
Strumenti per la creazione di dati	32

Pacchetto WebSphere Commerce Loader.	32
Servizi del negozio	32
Console di gestione.	32
WebSphere Commerce Accelerator.	32
Console di gestione dell'organizzazione	32
Grafico di riepilogo degli strumenti e dei dati del negozio.	32

Parte 4. Sviluppo dei dati del negozio 37

Capitolo 5. Risorse del sito 39

Risorse del sito in WebSphere Commerce	39
Lingua	39
Attributi membro	40
Tipi di attributi	40
Tipi di gruppi di membri.	40
Utente	40
Organizzazione	40
Ruolo	40
Conversione unità di quantità	41
Unità di quantità	41
Tipi di tasse	41
Utilizzi del calcolo	41
Valuta	41
Utilizzo numero	41
Tipi di articoli	41
Formati unità.	41
Creazione di risorse del sito in WebSphere Commerce.	42

Capitolo 6. Risorse del negozio 43

Risorse del negozio in WebSphere Commerce	43
Entità negozio	44
Creazione di risorse del negozio in WebSphere Commerce.	44
Creazione di risorse dati del negozio in un file XML.	45

Capitolo 7. Dati di registro comandi, viste e URL. 51

Registrazione di comandi, viste e URL in WebSphere Commerce.	51
Creazione di un file XML per registrare comandi, viste e URL	51

Capitolo 8. Risorse catalogo 55

Cataloghi in WebSphere Commerce	55
Cataloghi	56
Gruppi di catalogo	56
Voci di catalogo	57
Prodotti	57
Articoli.	57
Pacchetti	57

Bundle	58
Kit dinamici	58
Gruppi di prodotti	58
Attributi	58
Valori di attributo	58
Attributi di pacchetto	59
Valori degli attributi di pacchetto	59
Creazione di risorse catalogo in WebSphere Commerce	59
Creazione di un catalogo principale	60
Visualizzazione delle risorse catalogo del negozio	77
Creazione del catalogo di navigazione	80
Creazione dei cicli di categorie	80
Aggiunta di un prodotto a una seconda categoria	82
Creazione delle risorse catalogo in WebSphere Commerce	84
Gruppi di catalogo	85
Voci di catalogo	86
Strumenti di gestione prodotti	87
Catalog Manager	88
Capitolo 9. Risorse assegnazione prezzi	89
Assegnazione prezzi in WebSphere Commerce	89
Offerta	89
Prezzo offerta	90
Contenitore di posizioni commerciali	90
Termini e condizioni	90
Termini e condizioni per l'attribuzione prezzi	90
Accordo commerciale	90
Partecipante	91
Ruolo partecipante	91
Contratto	91
Politica aziendale	91
Politica prezzi	92
Spedizione delle voci di catalogo	92
Altre risorse assegnazione prezzi	92
Creazione di risorse assegnazione prezzi in WebSphere Commerce	92
Creazione di risorse assegnazione prezzi in un file XML	93
Capitolo 10. Risorse contratti	97
Contratti in WebSphere Commerce	98
Conto (conto commerciale)	98
Contratto	99
Accordo commerciale	99
Termini e condizioni	100
Politiche aziendali	100
Allegato	100
Articolo dell'ordine	100
Creazione di una risorsa contratto predefinita in WebSphere Commerce	101
Creazione dei file XML di politica aziendale	103
Creazione di un file XML di contratti predefiniti	104
Capitolo 11. Risorse evasione ordini	109
Risorse evasione ordini in WebSphere Commerce	110
Centro di evasione ordini	110
Ricevute	111

RaDetail	111
Inventario	111
Disposizioni sulla spedizione	111
Altre risorse evasione ordini	111
Creazione di risorse evasione ordini in WebSphere Commerce	112
Creazione di risorse evasione ordini del negozio	114

Capitolo 12. Risorse Campagne 117

Campagne in WebSphere Commerce	117
Creazione di risorse campagne in WebSphere Commerce	118

Capitolo 13. Risorse pagamento 119

Creazione di risorse pagamento mediante un file XML	119
---	-----

Capitolo 14. Risorse lingua 121

Risorse lingua in WebSphere Commerce	121
Lingua predefinita	121
Lingua supportata	122
Lingua alternativa	122
Creazione di risorse lingua in WebSphere Commerce	122

Capitolo 15. Risorse valuta 123

Risorse valuta in WebSphere Commerce	123
Formato valuta	124
Utilizzo numero	124
Descrizione del formato valuta	124
Valuta supportata	124
Regole di conversione valute	124
Valuta di controvalore	124
Creazione di risorse valuta in WebSphere Commerce	125
Creazione di risorse valuta mediante un file XML	125

Capitolo 16. Risorse unità di misura 129

Unità di misura in WebSphere Commerce	129
Unità di quantità e formato dell'unità di quantità	130
Creazione di unità di misura in WebSphere Commerce	131

Capitolo 17. Risorse giurisdizioni 133

Risorse giurisdizioni in WebSphere Commerce	133
Creazione di risorse giurisdizioni in WebSphere Commerce	134

Capitolo 18. Risorse spedizione 135

Risorse spedizione in WebSphere Commerce	135
Modalità di spedizione	136
Codici di calcolo	136
Giurisdizioni e gruppi di giurisdizioni	137
Creazione di risorse spedizione in WebSphere Commerce	137
Creazione di risorse spedizione mediante un file XML	138

Creazione di risorse evasione ordini per la spedizione	146
Creazione di risorse spedizione catalogo del negozio	149
Creazione di una modalità di spedizione predefinita	150
Capitolo 19. Risorse tasse	153
Risorse tasse in WebSphere Commerce	153
Categorie di tasse	153
Codice di calcolo	154
Giurisdizioni e gruppi di giurisdizioni	155
Creazione di risorse tasse in WebSphere Commerce	156
Creazione di risorse tasse mediante un file XML	156
Creazione di risorse evasione ordini per le tasse	167
Creazione di risorse tasse catalogo del negozio	168
Capitolo 20. Risorse sconti	171
Sconti in WebSphere Commerce	171
Codice di calcolo	171
Creazione di risorse sconti in WebSphere Commerce	172
Capitolo 21. Risorse inventario	173
Risorse inventario in WebSphere Commerce	173
Inventario ATP	174
Inventario non ATP	175
Creazione di risorse inventario in WebSphere Commerce	176
Capitolo 22. Risorse ordini	177
Risorse ordini in WebSphere Commerce	177
Ordini e articoli degli ordini	177
Articoli ordine	178
Creazione di risorse ordine in WebSphere Commerce	180
Capitolo 23. Risorse clienti e rivenditori	181
Risorse clienti in WebSphere Commerce	181
Informazioni sull'indirizzo	182
Elenchi di interessi	182
Comprensione delle risorse del venditore in WebSphere Commerce	182
Negozi	183
Conti	184
Contratti	184
Gruppi di prodotti	184
Listino prezzi	184
Cataloghi	185
Centri di evasione ordini	185
Articoli di inventario	185
Risorse membri in WebSphere Commerce	186
Membri	187
Attributi membro	187
Ruoli	188
Creazione di risorse membri in WebSphere Commerce	188

Parte 5. Aggiunta del controllo accessi al negozio 189

Capitolo 24. Controllo dell'accesso al negozio 191

Comprensione del controllo accessi in WebSphere Commerce	191
Politiche di controllo accessi	191
Il controllo accessi nei negozi	193
Aggiunta del controllo accessi al negozio	198
Creazione del controllo accessi nel negozio	198
Modifica dei file di controllo accessi nell'archivio di esempio	201

Parte 6. Organizzazione del negozio 205

Capitolo 25. Organizzazione di un negozio 207

Creazione di un archivio del negozio	207
Creazione di un archivio negozio di esempio	209

Parte 7. Pubblicazione del negozio 213

Capitolo 26. Pubblicazione di un negozio completo 215

Comprensione della pubblicazione in WebSphere Commerce	215
Avvio della pubblicazione	216
Verifiche preliminari alla pubblicazione.	217
Pubblicazione delle risorse	219
Configurazione del pagamento	226
File di log della pubblicazione.	226

Capitolo 27. Panoramica sul caricamento dei dati 229

Comprensione del caricamento dati in WebSphere Commerce	230
Comandi del pacchetto Loader per il caricamento dei dati del negozio	234
Comandi del pacchetto Loader per la trasformazione e l'estrazione dei dati	250
Strumenti relativi ai comandi del pacchetto Loader	259
Caricamento dei dati del negozio.	260
Uso dei comandi e degli script del pacchetto Loader	261
Esempi di risoluzione degli identificativi	262
Esempio di caricamento dei dati	269

Capitolo 28. Caricamento dei gruppi di risorse di database di WebSphere Commerce 271

Gruppi di risorse database	271
Sequenza di caricamento delle risorse database	271
Caricamento di un negozio.	273

Caricamento dei gruppi di risorse database 279

Capitolo 29. Pubblicazione dei contratti e dei conti commerciali 285

Pubblicazione di contratti e conti commerciali con Servizi del negozio o la riga comandi 286
Pubblicazione di contratti e conti commerciali mediante i comandi 286
 Pubblicazione di risorse conto commerciale 286
 Pubblicazione delle risorse contratto. 287

Capitolo 30. Pubblicazione delle risorse della facciata e dei file di configurazione del negozio 289

Pubblicazione delle risorse della facciata e dei file di configurazione del negozio mediante Servizi del negozio o la riga comandi 289
Pubblicazione dei file di configurazione e delle risorse facciata del negozio mediante copia su WebSphere Commerce Server 290

Parte 8. Aggiunta delle funzioni di WebSphere Commerce al negozio . 293

Capitolo 31. Aggiunta dell'assistenza clienti a un negozio. 295

Comprensione della funzione di assistenza clienti in un negozio 295
 Uso del frameset 296
 Controllo dei clienti mediante l'assistenza clienti 298
Aggiunta dell'assistenza clienti a un negozio 302
 Parte 1: Prerequisiti per l'installazione 302
 Parte 2: Copia dei file di integrazione dell'assistenza clienti dal negozio di esempio. . . . 302
 Parte 3: Aggiunta del frameset al negozio 303
 Parte 4: Aggiunta del codice per ottenere l'ID o il nome del cliente. 304

Parte 5: Aggiunta del codice per determinare quale pagina sta sfogliando il cliente 305

Parte 6: Aggiunta del codice per tenere traccia del numero di articoli nel carrello degli acquisti . 306

Parte 7: Aggiunta di un collegamento all'assistenza clienti 306

Parte 8: Modifica dei messaggi che vengono visualizzati al cliente 307

Capitolo 32. Aggiunta di spot di e-Marketing al negozio 309

Spot di e-Marketing 309
 Bean e-MarketingSpot 312
Aggiunta di uno spot di e-Marketing alle pagine del negozio 312

Parte 9. Appendici 315

Appendice A. Legenda UML 317

Appendice B. Creazione di dati. 319

Creazione di dati per i negozi di esempio 319
 Servizi del negozio e negozi di esempio 320

Appendice C. sarinfo.xml 321

Esempio di sarinfo.xml 321

Appendice D. sarrule.xml 327

Esempio di sarrule.xml 327

Appendice E. Gruppi di risorse database 331

Dipendenze di gruppi di risorse database 331

Appendice F. Informazioni particolari 337

Marchi 339

Prima di iniziare

La Guida per lo sviluppatore del negozio *IBM WebSphere Commerce* fornisce informazioni sull'architettura del negozio di WebSphere Commerce e sul processo di sviluppo del negozio. In particolare, contiene dettagli relativi ai seguenti argomenti:

- Opzioni per lo sviluppo del negozio
- Archivio del negozio
- Strumenti di sviluppo del negozio
- Sviluppo della facciata del negozio
- Sviluppo dei dati del negozio
- Architettura dati del negozio
- Modello di informazioni dati del negozio
- Aggiunta del controllo accessi al negozio
- Organizzazione del negozio
- Pubblicazione del negozio
- Aggiunta delle funzioni di WebSphere Commerce al negozio

Convenzioni utilizzate in questo manuale

Questa pubblicazione utilizza le seguenti convenzioni tipografiche:

Grassetto: indica i comandi o i controlli della GUI (Graphical User Interface) come i nomi dei campi, i pulsanti o le scelte di menu.

Monospazio: indica gli esempi di testo da immettere esattamente come sono mostrati ed i percorsi delle directory.

Corsivo: utilizzato per evidenziare delle parti e per indicare le variabili cui l'utente attribuisce dei valori.



Questa icona indica un suggerimento, vale a dire ulteriori informazioni che aiutano a completare un'attività.

NT indica le informazioni specifiche per Windows NT.

2000 indica le informazioni specifiche per Windows 2000.

AIX indica le informazioni specifiche per AIX.

Solaris indica le informazioni specifiche per il sistema operativo Solaris.

400 indica le informazioni specifiche per IBM eserver iSeries 400 (precedentemente noto come AS/400).

Linux indica le informazioni specifiche per Linux.

Business indica le informazioni specifiche per WebSphere Commerce Business Edition.

Professional indica le informazioni specifiche per WebSphere Commerce Professional Edition.

Dove reperire ulteriori informazioni

Questo manuale potrà essere aggiornato in futuro. Per aggiornamenti, visitare il sito Web di WebSphere Commerce al seguente indirizzo:

 http://www.ibm.com/software/webservers/commerce/wc_be/lit-tech-general.html

 http://www.ibm.com/software/webservers/commerce/wc_pe/lit-tech-general.html

Gli aggiornamenti possono includere nuove informazioni.

Parte 1. Panoramica

Capitolo 1. Panoramica sull'architettura del negozio

In questo capitolo viene fornita un'introduzione all'architettura del negozio di WebSphere Commerce Server.

Che cos'è un negozio in linea?

Un negozio in linea è un negozio che si serve delle tecnologie Internet per la vendita o lo scambio di merci o servizi. Comprende una serie di pagine Web che visualizzano i propri prodotti e consentono ai clienti di acquistarli. La home page porta i clienti nel negozio, dirigendoli verso i prodotti e i servizi. Le pagine di catalogo in linea raggruppano i prodotti ed indirizzano i clienti alle pagine di prodotti specifici, dove possono trovare informazioni dettagliate relative al prodotto. Nei negozi business-to-consumer, la pagina del carrello degli acquisti ha le stesse caratteristiche di un carrello reale: è possibile aggiungervi i prodotti da acquistare, che verranno pagati nelle pagine di Verifica. Nei siti business-to-business, determinate pagine permettono di inoltrare ordini e RFQ (Request For Quotes).

Le pagine del negozio vengono create utilizzando la tecnologia JPS. Ciascuna pagina contiene HTML per contenuto statico, JavaScript lato client per gestire dati di immissione e visualizzazione dati sofisticati, URL per richiamare i comandi WebSphere Commerce Server ed altre visualizzazioni e tag JSP e codici Java per creazioni di contenuto dinamico. Un insieme di bean di dati commerciali forniti con WebSphere Commerce Studio e WebSphere Commerce sono disponibili per essere utilizzati dai file JavaServer Page e consentono di accedere alle informazioni del database, ad esempio al prezzo di un prodotto o agli attributi del prodotto.

Il negozio è composto anche da risorse database necessarie per creare un negozio funzionale. Ad esempio, un negozio funzionale deve includere i dati relativi ai cataloghi, alle tasse, alla spedizione e alla valuta.

Composizione di un negozio

Un negozio in linea è composto dalle seguenti risorse:

- Facciata del negozio

La parte esterna del negozio o la parte che viene visualizzata ai clienti è detta facciata del negozio. La facciata del negozio comprende risorse Web come pagine HTML, file JSP, fogli di stile, immagini, grafici e altri tipi di file multimediali.

Questa guida illustra i concetti e le attività relativi alla creazione di file JSP che generano le pagine del negozio. Per ulteriori informazioni, consultare Capitolo 3, "Sviluppo della facciata del negozio" a pagina 17.

- Ufficio posteriore

La parte del negozio non visibile al cliente, i comandi, il codice personalizzato e l'implementazione della logica aziendale che consentono ad un cliente di acquistare un prodotto dalla facciata del negozio, è detta ufficio posteriore.

Per informazioni più dettagliate sulla creazione della logica aziendale o del codice personalizzato, fare riferimento a *IBM WebSphere Commerce - Guida per il programmatore*.

- Dati del negozio

Le risorse dati che compongono il negozio. Per funzionare correttamente, un negozio deve avere i dati necessari per supportare tutte le attività dei clienti. Ad esempio, perché un cliente possa fare un acquisto, il negozio deve presentare un catalogo delle merci in vendita, incluso un processo per la gestione degli ordini, un inventario per la compilazione della richiesta e un processo di consegna. Deve presentare, inoltre, metodi per l'elaborazione e la raccolta dei pagamenti. I concetti e le attività relativi alla creazione dei dati del negozio sono esaminati nella Parte 4, "Sviluppo dei dati del negozio" a pagina 37.

Architettura del negozio

L'architettura del negozio di WebSphere Commerce è formata dai seguenti componenti:

- WebSphere Commerce Server
- Istanza di WebSphere Commerce Server
- Configurazioni del negozio

WebSphere Commerce Server

WebSphere Commerce Server è il server che gestisce le funzioni relative al commercio e al negozio di una soluzione di commercio elettronico. Le risorse della facciata del negozio e la logica aziendale dell'ufficio posteriore risiedono in un'applicazione Web all'interno di WebSphere Commerce Server. WebSphere Commerce fornisce un'applicazione Web predefinita (WCS Stores) ma è possibile anche crearne una personalizzata.

Un'applicazione Web può contenere le risorse relative a un unico negozio o a più negozi. Quando un'applicazione Web contiene più facciate di negozi o uffici posteriori, le risorse relative a ciascun negozio sono separate dalla directory del negozio (dirnegozio).

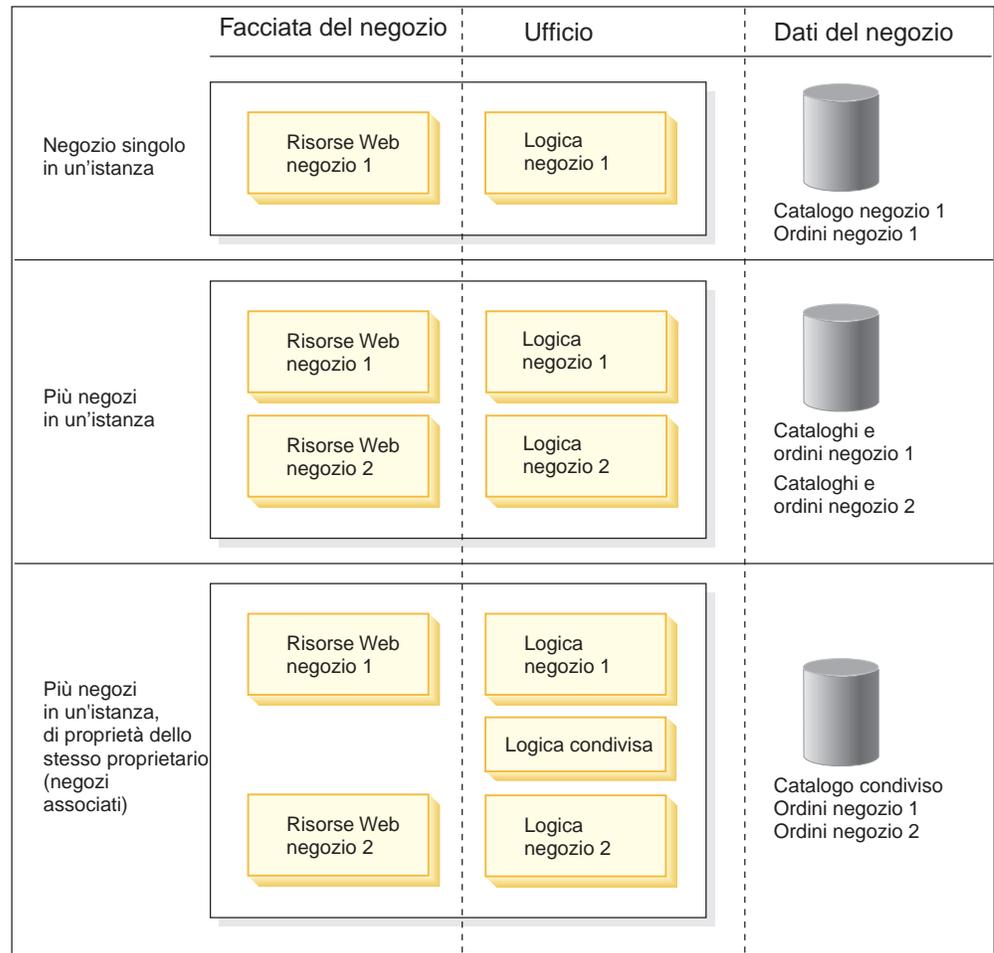
Istanza di WebSphere Commerce Server

Un'istanza di WebSphere Commerce Server è un'applicazione WebSphere Application Server con un database associato. Un'istanza può supportare più negozi. Tutti i negozi di un'istanza condividono lo stesso database e possono condividere alcuni tipi di dati, ad esempio cataloghi, evasione ordini o ricevute. Tutti i negozi di un'istanza condividono lo stesso contenitore EJB.

Configurazioni del negozio

WebSphere Commerce supporta varie configurazioni del negozio. Ciò significa che, mediante WebSphere Commerce, è possibile creare in un'istanza un unico negozio oppure più negozi con facciate, uffici posteriori e dati diversi. In alternativa, è possibile creare in un'istanza più negozi con facciate diverse e uffici posteriori e cataloghi condivisi. Il grafico di seguito riportato descrive alcune possibili

configurazioni del negozio.



Nota: Ogni negozio ha un proprio identificativo. Le licenze per WebSphere Commerce impostano un limite specifico per il numero di negozi che è possibile creare. E' possibile acquistare autorizzazioni supplementari. Per ulteriori informazioni, consultare la sezione sugli accordi di licenza.

Capitolo 2. Sviluppo del negozio

In questo capitolo viene fornita una panoramica del processo di sviluppo del negozio in WebSphere Commerce.

Opzioni per lo sviluppo del negozio

WebSphere Commerce fornisce varie opzioni per lo sviluppo del negozio:

- Creazione di un negozio basato su un esempio
- Creazione di un negozio sviluppando nuove risorse del negozio
- Creazione di un negozio utilizzando una combinazione del negozio di esempio e delle nuove risorse del negozio

Creazione di un negozio basato su un esempio

In WebSphere Commerce, il modo più facile e veloce per creare un negozio in linea è quello di copiare uno dei negozi di esempio forniti con WebSphere Commerce e di personalizzare il negozio in base alle proprie esigenze.

Negozi di esempio

WebSphere Commerce comprende diversi negozi di esempio in linea perfettamente funzionali che possono essere utilizzati come base per la creazione del negozio. Tali esempi, che comprendono sia i negozi business-to-consumer che i negozi business-to-business, implementano la maggior parte delle funzioni comunemente utilizzate nei principali siti di commercio elettronico del momento e forniscono tutte le risorse del negozio necessarie al funzionamento del negozio stesso. Per ulteriori informazioni sui negozi in linea forniti con WebSphere Commerce, fare riferimento alla guida in linea di WebSphere Commerce.

Perché partire con un negozio di esempio: Nonostante sia possibile creare un negozio in linea con WebSphere Commerce, se si utilizza una copia di uno dei negozi di esempio come base, sarà possibile creare un negozio funzionale molto più rapidamente.

Per creare un negozio funzionale, WebSphere Commerce richiede che alcuni dati vengano caricati nel database di WebSphere Commerce Server e che tali dati vengano caricati secondo lo schema in un preciso ordine. Poiché i negozi di esempio comprendono tutti i dati obbligatori nell'ordine e nella struttura richiesta dal database di WebSphere Commerce Server, se si utilizza un negozio di esempio come base per il negozio, viene ridotto notevolmente il periodo per la creazione iniziale.

Una volta creata una copia di un negozio di esempio, è possibile modificarlo completamente o in parte in base alle necessità del negozio. Ad esempio, è possibile limitarsi a modificare i dati mediante gli strumenti disponibili con WebSphere Commerce e l'aspetto delle pagine del negozio mediante WebSphere Commerce Studio. In alternativa, è possibile modificare i file XML o direttamente il database per apportare variazioni più complete ai dati e riscrivere le pagine del negozio per modificare il flusso e le funzioni del negozio stesso. Per ulteriori informazioni sulla modifica dei negozi, fare riferimento alla sezione *"Modifica delle risorse database del negozio"* della guida in linea di WebSphere Commerce.

WebSphere Commerce fornisce inoltre vari negozi di riferimento progettati per essere utilizzati come esempi di codice per particolari funzioni. Un negozio di riferimento è un negozio in linea che contiene un codice funzionale completo per le funzioni selezionate di un negozio in linea, ad esempio le aste. I negozi di riferimento sono disponibili all'indirizzo

 http://www.ibm.com/software/webservers/commerce/wc_be/downloads.html

 http://www.ibm.com/software/webservers/commerce/wc_pe/downloads.html

Per ulteriori informazioni sulla creazione di un negozio basato su un esempio, fare riferimento alla guida in linea di WebSphere Commerce.

Creazione di un negozio sviluppando nuove risorse del negozio

Non tutti desiderano creare il negozio in base a un negozio di esempio. Ad esempio, se il flusso delle pagine del negozio è molto diverso da quello dei negozi di esempio forniti o se si desidera personalizzare lo schema del database di WebSphere Commerce Server, è possibile creare il negozio sviluppando la facciata del negozio, l'ufficio posteriore e le risorse dati. Per un elenco degli strumenti forniti con WebSphere Commerce, fare riferimento alla sezione "Strumenti di sviluppo del negozio" a pagina 10.

Creazione di un negozio utilizzando una combinazione del negozio di esempio e delle nuove risorse del negozio

L'utilizzo di una combinazione del negozio di esempio e delle nuove risorse del negozio può rappresentare il metodo di sviluppo che maggiormente soddisfa le esigenze del cliente. Ad esempio, se alcune risorse database di uno dei negozi di esempio rispondono esattamente alle esigenze del negozio, ma il flusso delle pagine di quel negozio no, è possibile copiare le risorse database e personalizzarle sviluppando al tempo stesso risorse Web completamente nuove.

L'archivio negozio

I negozi di esempio inclusi in WebSphere Commerce sono forniti in un formato di archivio negozio. Un file di archivio negozio (.sar) è un file di archivio compresso (ad esempio un file ZIP) contenente tutte le risorse necessarie per la creazione di un negozio. Viene utilizzato principalmente come veicolo per la preparazione e la distribuzione di negozi in un formato facile da copiare, quindi come base su cui creare nuovi negozi. Per creare un negozio funzionale da visualizzare, sfogliare e in cui fare shopping, un archivio negozio deve semplicemente essere pubblicato su WebSphere Commerce Server.

In genere, un archivio negozio è formato dai seguenti file:

- Risorse Web: I file che creano le pagine del negozio, quali file HTML, file JSP, immagini, grafici e file di inclusione. Le risorse Web sono raggruppate in un file compresso nell'archivio negozio.
- Bundle di risorse proprietà (facoltative): Contengono il testo per le pagine del negozio. Se il negozio supporta più di una lingua, il bundle di risorse conterrà più bundle, uno per lingua.
- Risorse dati del negozio: I dati da caricare nel database. Le risorse dati del negozio includono dati quali informazioni sulle campagne, le voci di catalogo, le valute, l'evasione ordini e informazioni sull'assegnazione dei prezzi, la

spedizione, il negozio e la tassazione. Per un elenco più dettagliato delle risorse dati del negozio, fare riferimento alla Parte 4, “Sviluppo dei dati del negozio” a pagina 37.

Le risorse database del negozio negli archivi negozio di esempio fornite con WebSphere Commerce assumono la forma di file XML validi per il Pacchetto Loader. I file XML dell’archivio negozio sono concepiti per essere trasportati e non possono contenere chiavi primarie generate specifiche per una particolare istanza del database. Essi utilizzano invece alias interni che vengono risolti da ID Resolver al momento della pubblicazione. L’utilizzo di queste convenzioni permette agli archivi negozio di esempio di essere copiati e pubblicati più volte. Per ulteriori informazioni, consultare l’Appendice B, “Creazione di dati” a pagina 319.

Per ulteriori informazioni sul pacchetto Loader, consultare la Parte 7, “Pubblicazione del negozio” a pagina 213.

- Risorse pagamento: Informazioni sulla configurazione per IBM Payment Manager.
- Descrittore: Un file XML, `sarinfo.xml`, che descrive l’archivio negozio, inclusi i nomi dei file di archivio compressi delle risorse Web, i bundle di risorse e i file XML delle risorse database del negozio. Il file `sarinfo.xml` contiene anche i nomi dei file di inclusione e dei file di controllo congruenza, nonché le informazioni sul file di archivio necessarie durante il processo di pubblicazione. Il file `sarinfo.xml` è l’unico file obbligatorio in un archivio negozio.

Nota: Gli archivi dei negozi di esempio  ToolTech e NewFashion includono anche i seguenti file:

- `tools_properties.zip`
- `tools_xml.zip`
- `runtime_xml.zip`

Questi file vengono utilizzati da Servizi del negozio per la configurazione dei negozi. Tali file non devono essere modificati, rimossi o copiati in altri negozi.

Archivi negozio di esempio

Un file dell’archivio negozio di esempio (`.sar`) rappresenta un archivio negozio da copiare ed utilizzare come una base su cui creare nuovi negozi. Gli archivi negozio di esempio includono poche convenzioni che consentono di copiare e pubblicare diversi articoli. Queste convenzioni comprendono quanto segue:

- Nessun riferimento alle chiavi primarie o a quelle esterne generate: Gli archivi negozio di esempio non contengono le chiavi primarie specifiche di una determinata istanza del database. Essi utilizzano invece alias interni che vengono risolti da ID Resolver al momento della pubblicazione. Per ulteriori informazioni, consultare l’Appendice B, “Creazione di dati” a pagina 319.

I negozi di esempio forniti con WebSphere Commerce sono archivi di negozi di esempio. Questi negozi sono disponibili nell’elenco **Esempi** della pagina di creazione dell’archivio negozio di Servizi del negozio.

Quando utilizzare un archivio negozio

L’archivio negozio è concepito come un veicolo per la preparazione e la distribuzione di altri negozi. Se si desidera utilizzare il negozio come esempio,

sviluppare il negozio su un altro server o piattaforma oppure utilizzarlo come base per la creazione di altri negozi, è possibile organizzare il negozio nel formato di archivio negozio.

E' inoltre possibile utilizzare l'archivio negozio se il negozio è strettamente basato su uno dei negozi di esempio e non si ha alcuna necessità di apportare modifiche sostanziali all'archivio.

Se si sceglie di utilizzare l'archivio negozio ed il negozio è stato creato mediante un negozio di esempio, il negozio già sarà nel formato di archivio negozio. E' possibile lasciare il nuovo negozio nel formato di archivio negozio verificando che tutte le modifiche effettuate durante lo sviluppo del negozio siano riportate nell'archivio negozio.

E' inoltre possibile organizzare come archivio negozio un negozio creato con altri metodi. Per ulteriori informazioni sulla creazione di un archivio negozio, fare riferimento alla Parte 6, "Organizzazione del negozio" a pagina 205.

Quando non utilizzare un archivio negozio

E' possibile stabilire di non utilizzare un archivio negozio se si desidera creare un'unica istanza del negozio oppure se si desidera apportare modifiche significative allo schema esistente di WebSphere Commerce. Le modifiche allo schema di WebSphere Commerce non sono supportate dall'archivio negozio e dagli strumenti di sviluppo del negozio di WebSphere Commerce non contenuti nella finestra. Se si desidera conservare l'archivio negozio dopo aver apportato modifiche significative allo schema, contattare un rappresentante IBM per ulteriori informazioni.

Strumenti di sviluppo del negozio

WebSphere Commerce fornisce una varietà di strumenti per facilitare lo sviluppo del negozio. Gli strumenti da utilizzare dipendono dal modo in cui si sceglie di sviluppare e organizzare il negozio.

Strumenti per lo sviluppo della facciata del negozio

Lo sviluppo delle risorse della facciata del negozio può includere la personalizzazione delle pagine del negozio di esempio, la sostituzione di tali pagine con pagine già esistenti, la creazione di nuove pagine e la combinazione di questi tre processi.

WebSphere Commerce fornisce i seguenti strumenti per la creazione o la modifica delle risorse della facciata del negozio:

- **WebSphere Commerce Studio:** Commerce Studio include gli strumenti necessari per creare e modificare le risorse della facciata del negozio, inclusi file HTML, grafici, file multimediali e file JSP (JavaServer Pages). Page Designer, incluso in Commerce Studio, consente di creare file HTML o JSP e immagini animate. È tuttavia possibile configurare WebSphere Commerce Studio per l'utilizzo di un altro strumento di sviluppo Web a scelta. Per ulteriori informazioni sulla registrazione degli strumenti con WebSphere Commerce Studio, fare riferimento alla guida in linea di WebSphere Studio.

Se si prevede di utilizzare il negozio nel formato di archivio negozio, WebSphere Commerce Studio consente di importare le risorse Web dall'archivio negozio in un progetto di Studio, mantenendo intatta la struttura dell'archivio negozio. Una volta apportate le modifiche ai file JSP, ai file HTML e alle immagini utilizzando

gli strumenti di Studio, è possibile esportare i file nuovamente nell'archivio negozio su WebSphere Commerce Server e ripubblicare le risorse Web.

Se non si desidera conservare l'archivio negozio, è possibile pubblicare i file direttamente nel negozio funzionale mediante WebSphere Commerce Studio.

- **Servizi del negozio:** La finestra Risorse Web in Servizi del negozio consente di sostituire il file ZIP delle risorse Web nell'archivio negozio con un altro insieme di risorse Web oppure di scaricare le risorse Web esistenti nella posizione scelta, da cui è possibile modificarle mediante lo strumento di sviluppo Web preferito. Se si sta utilizzando il negozio nel formato di archivio negozio, è possibile aprire la finestra Risorse Web per riportare le risorse modificate nell'archivio negozio. La pagine di configurazione dei negozi consentono di abilitare o disabilitare le diverse funzioni dei file JSP nei negozi pubblicati. Attualmente, lo strumento Servizi del negozio supporta solo la configurazione delle funzioni di collaborazione: Collaborative Workspaces e Customer Care. Queste funzioni sono disponibili per la configurazione solo nei negozi basati sui negozi di esempio  ToolTech e NewFashion.

Per ulteriori informazioni sull'uso degli strumenti WebSphere Commerce Studio e Servizi del negozio per la creazione e la modifica delle risorse della facciata del negozio, fare riferimento alla guida in linea di WebSphere Commerce. Per ulteriori informazioni sulla creazione della facciata del negozio in WebSphere Commerce, fare riferimento alla Parte 2, "Sviluppo della facciata del negozio" a pagina 15.

Strumenti per lo sviluppo dei dati del negozio

Esistono varie opzioni per lo sviluppo e la modifica delle risorse database del negozio.

- **Servizi del negozio**

Servizi del negozio è un insieme di strumenti per browser che funziona negli archivi negozi. Mediante Servizi del negozio, è possibile creare rapidamente un archivio negozio basato su un esempio fornito con WebSphere Commerce. Dopo aver creato un archivio negozio, Servizi del negozio consente di effettuare le seguenti operazioni:

- Pubblicazione dell'archivio negozio per creare un negozio funzionale.
- Modifica delle impostazioni di tassazione utilizzando il blocco appunti Tasse.
- Modifica delle impostazioni di spedizione utilizzando il blocco appunti Spedizione.
- Modifica delle impostazioni del negozio generali utilizzando il blocco appunti Profilo negozio.

Servizi del negozio non consente di modificare le risorse dati del negozio nell'archivio negozio. Per un elenco delle risorse che è possibile modificare mediante Servizi del negozio, fare riferimento alla sezione "*Modifica delle risorse database del negozio*" della guida in linea di WebSphere Commerce. Per modificare altre risorse nell'archivio negozio, modificare direttamente le risorse XML.

Per ulteriori informazioni sull'uso di Servizi del negozio, fare riferimento alla guida in linea di WebSphere Commerce.

Quando utilizzare Servizi del negozio: Utilizzare Servizi del negozio per copiare gli archivi negozio di esempio e modificare le risorse database nel formato di archivio negozio.

- Pacchetto WebSphere Commerce Loader

Il Pacchetto WebSphere Commerce Loader è composto principalmente da programmi di utilità per la preparazione e il caricamento dei dati in un database di WebSphere Commerce. Utilizzare il pacchetto Loader per caricare una grossa quantità di dati e per aggiornare i dati nel database di WebSphere Commerce. Il programma di utilità Loader di questo pacchetto utilizza file XML validi e in formato corretto come immissione per caricare i dati nel database. Gli elementi del documento XML vengono associati ai nomi della tabella nel database e gli attributi dell'elemento vengono associati alle colonne.

Per informazioni sull'uso del Pacchetto Loader per lo sviluppo e il caricamento delle risorse dati, fare riferimento alla Parte 7, "Pubblicazione del negozio" a pagina 213.

Quando utilizzare il Pacchetto WebSphere Commerce Loader: Utilizzare il Pacchetto WebSphere Commerce Loader per caricare le risorse database nel database di WebSphere Commerce ed aggiornarle.

Importante: Se lo schema del database è stato modificato, l'uso del Pacchetto Loader rappresenta l'unico modo per caricare i dati nel database.

- WebSphere Commerce Accelerator

WebSphere Commerce Accelerator rappresenta un workbench di strumenti in linea utilizzato principalmente per gestire i negozi in linea attraverso le varie operazioni del negozio. Tuttavia, poiché WebSphere Commerce Accelerator consente di modificare i dati già presenti nel database, una volta che il database è stato formato, è possibile utilizzarlo come uno strumento di sviluppo del negozio, se contiene i dati del negozio di esempio o i dati creati. Per un elenco delle risorse database che è possibile modificare con WebSphere Commerce Accelerator, fare riferimento alla guida in linea di WebSphere Commerce, alla sezione "Modifica delle risorse database del negozio".

Quando utilizzare WebSphere Commerce Accelerator: Utilizzare WebSphere Commerce Accelerator dopo aver formato il database di WebSphere Commerce.

- Modifica del database direttamente

E' sempre possibile modificare il database direttamente mediante inserimenti SQL.

Nota: SQL è un database specifico. Oracle può richiedere una sintassi SQL diversa. Le istruzioni SQL devono avere necessariamente valori specifici per il database e non possono essere riutilizzate in altre istanze di WebSphere Commerce Server.

Strumenti per lo sviluppo dell'ufficio posteriore

Gli strumenti per lo sviluppo dell'ufficio posteriore, inclusa la creazione e l'estensione dei comandi, la creazione del codice personalizzato e l'implementazione della logica aziendale sono esaminati in *IBM WebSphere Commerce - Guida per il programmatore*.

Il ruolo dello sviluppatore del negozio

Gli sviluppatori di negozio sviluppano tutti e tre i tipi di risorse del negozio. Progettano e realizzano le risorse della facciata del negozio, inclusi i file JavaServer Pages e le risorse dell'ufficio posteriore, compresa la creazione di nuovi comandi e di tutti i codici personalizzati necessari. Creano inoltre i dati del negozio e possono modificare le funzioni standard incluse in WebSphere Commerce.

Gli sviluppatori di negozio che creano la facciata del negozio e i dati del negozio devono avere esperienza nella programmazione in Java, JavaScript, HTML, tecnologia JSP e familiarità con l'architettura, i dati e gli archivi negozio di WebSphere Commerce.

Gli sviluppatori di negozio che creano l'ufficio posteriore devono avere esperienza nella programmazione in Java, JavaBeans, VisualAge per Java, programmazione J2EE e familiarità con il modello di programmazione e con il modello di oggetto di WebSphere Commerce. Il manuale *IBM WebSphere Commerce - Guida per il programmatore* fornisce ulteriori informazioni sulla personalizzazione dell'ufficio posteriore.

Gli sviluppatori di negozio possono collaborare con gli sviluppatori di database e i progettisti Web. Gli sviluppatori di database modificano ed estendono lo schema del database di WebSphere Commerce allo scopo di implementare le funzioni del negozio personalizzate o di integrare le informazioni di database esistenti. Questo membro è di norma dotato di capacità di gestione dei database per DB2 o Oracle.

I designer Web creano l'immagine e l'aspetto del sito e lavorano con gli sviluppatori del negozio per creare le pagine del negozio. I designer Web devono essere esperti nell'utilizzo di strumenti multimediali, HTML e JavaScript, e devono avere una certa familiarità con la tecnologia JSP.

Nota: I ruoli dello sviluppatore di database e del progettista Web non sono definiti in WebSphere Commerce Server. Se necessario, agli sviluppatori di database e ai progettisti Web deve essere assegnato l'accesso dello sviluppatore di negozio.

Una volta creato un archivio negozio, lo sviluppatore del negozio dispone dell'autorizzazione per apportare modifiche al negozio manualmente o attraverso il blocco appunti Profilo negozio oppure quello relativo alle tasse e le spedizioni ma non dispone dell'autorizzazione di pubblicare l'archivio negozio in WebSphere Commerce Server.

Parte 2. Sviluppo della facciata del negozio

Capitolo 3. Sviluppo della facciata del negozio

Questo capitolo fornisce una panoramica sull'architettura della facciata del negozio di WebSphere Commerce e descrive il modo in cui la parte esterna del negozio, le risorse Web quali pagine HTML, file JSP, fogli di stile, immagini, grafici e altri tipi di file multimediali vengono visualizzate ai clienti.

Architettura della facciata del negozio

WebSphere Commerce utilizza un sistema di *comandi* e *viste* per mostrare ai clienti le risorse Web di una facciata del negozio.

- I *comandi* svolgono uno specifico processo aziendale, ad esempio aggiungere un prodotto al carrello degli acquisti, elaborare un ordine, aggiornare una rubrica acquirenti o visualizzare una specifica pagina di prodotto. Quando l'azione viene completata, il comando restituisce una vista.
- Le *viste* mostrano i risultati dei comandi e le azioni dell'utente, ossia presentano le pagine del negozio (file JSP) ai clienti. Per fare in modo che la vista richiami un file JSP, il nome file JSP deve essere registrato con la visualizzazione nella tabella del registro visualizzazioni (VIEWREG). Il file JSP corrispondente viene memorizzato con il nome file JSP nella sottodirectory (dirnegozio) relativa al negozio, al di sotto della directory principale dell'applicazione web WCS Stores.

I comandi e le viste vengono richiamati mediante URL. Ad esempio, quando un cliente fa clic su **Carrello degli acquisti** nel negozio di esempio, il cliente richiama l'URL `https://hostname/path/OrderItemDisplay?`, che viene trasferito in WebSphere Commerce Server. WebSphere Commerce Server richiama il comando `OrderItemDisplay` e la pagina del carrello degli acquisti viene visualizzata al cliente.

Quando un cliente fa clic su `?` nel negozio di esempio, il cliente richiama l'URL `https://hostname/path/HelpView?`, che viene trasferito in WebSphere Commerce Server. WebSphere Commerce Server richiama il comando `HelpView`, che restituisce la pagina della Guida.

WebSphere Commerce Server può inoltre creare una corrispondenza tra più comandi e un URL, il che consente a ciascun negozio di eseguire facoltativamente un determinato comando.

Allo stesso modo, WebSphere Commerce Server consente di creare una corrispondenza tra più file JSP ed un'unica visualizzazione, in cui ciascun negozio può facoltativamente registrare nomi file JSP diversi per i diversi tipi di unità.

Nota: I comandi di visualizzazione prodotti e categorie restituiscono viste e nomi file JSP. I nomi file JSP, che visualizzano prodotti e categorie, sono memorizzati nei dati del catalogo. Per ulteriori informazioni, consultare la sezione "Visualizzazione delle risorse catalogo del negozio" a pagina 77. E' possibile assegnare facoltativamente nomi file JSP diversi ai prodotti e alle categorie di visualizzazione per ciascun gruppo o lingua supportato dal negozio.

Comandi e viste predefiniti

WebSphere Commerce fornisce comandi e viste predefiniti che è possibile utilizzare nel negozio. I comandi e le viste predefiniti sono elencati nel file `wcs.bootstrap.xml`. I file bootstrap si trovano nelle seguenti directory:

-  unità:\WebSphere\CommerceServer\schema\xml
-  unità:\Program Files\WebSphere\CommerceServer\schema\xml
-  /usr/WebSphere/CommerceServer/schema/xml
-  /opt/WebSphere/CommerceServer/schema/xml
-  /opt/WebSphere/CommerceServer/schema/xml
-  /qibm/proddata/WebCommerce/schema/xml

Se manca un comando o una visualizzazione necessario, è possibile crearlo. Per informazioni sulla creazione dei comandi e delle viste, consultare il manuale *IBM WebSphere Commerce - Guida per il programmatore*.

Creazione di pagine del negozio

L'attività più complessa nella creazione della facciata del negozio è la creazione di pagine effettive del negozio. Prima di cominciare il lavoro di sviluppo relativo alle pagine del negozio, si consiglia di completare le seguenti attività di pianificazione:

- Sviluppo di un elenco di pagine del negozio necessarie
- Sviluppo di un elenco di URL di comandi e viste
- Associazione di nomi file JSP e viste

Sviluppo di un elenco di pagine del negozio

Per sviluppare un elenco di pagine necessarie per la creazione del negozio, è necessario conoscere i requisiti aziendali e funzionali del negozio e i processi aziendali che sono stati definiti.

Casi di utilizzo

Molti utenti stabiliscono i propri requisiti sotto forma di casi di utilizzo. I casi di utilizzo definiscono i processi aziendali del negozio, sotto forma di interazioni tra il cliente e il sistema proposto. Nel caso di un negozio in linea, i casi di utilizzo possono definire il modo in cui un cliente effettua la registrazione al negozio, sfoglia il catalogo oppure ordina un articolo.

Nella guida in linea viene fornita una serie di casi di utilizzo, che descrivono i processi aziendali relativi ai negozi di esempio. Questi casi di utilizzo consentono di semplificare la comprensione del flusso dei negozi di esempio e possono essere utilizzati come guida per la creazione di altri casi per il nuovo negozio.

Il seguente è un esempio di un elemento Registration:

Caso di utilizzo della registrazione: Il processo di registrazione consente ai clienti di immettere informazioni personali nel database.

Attore:

- Cliente

Flusso principale: Il cliente seleziona **Esegui la registrazione** dalla barra laterale. Il sistema visualizza una pagina con i seguenti campi:

- E-mail
- Password
- Verifica password
- Nome
- Cognome
- Et  (facoltativo)
- Sesso (facoltativo)

Il cliente immette le informazioni adeguate nei campi e seleziona **Inoltra**. Il sistema crea un nuovo cliente nel sistema e salva le informazioni del cliente (E1, E2, E3). Richiede quindi al cliente di gestire il conto secondo il processo descritto nel caso di utilizzo di gestione del conto personale

Flussi alternativi: Nessuno.

Flussi di eccezione: E1: L'indirizzo e-mail risulta gi  esistente:

- Se l'indirizzo e-mail risulta gi  presente nel sistema, verr  visualizzato un messaggio di errore, che richiede all'utente di fornire un altro indirizzo. La procedura ricomincia dall'inizio.

E2: Campi obbligatori mancanti:

- Se uno dei seguenti campi (E-mail, Password, Verifica password, Nome, Cognome)   vuoto, il sistema riporta un messaggio di errore. La procedura ricomincia dall'inizio.

E3: Password non valida:

- Se la password non   valida o non corrisponde alla password di verifica, il sistema invia un messaggio di avvertenza.

Flusso di acquisti del negozio: Indipendentemente dal fatto che per illustrare i processi aziendali del negozio vengano sviluppati casi di utilizzo o venga utilizzato un altro metodo, una volta che i processi aziendali sono disponibili,   possibile creare il flusso di acquisti del negozio.

Nota: Dal momento che i casi di utilizzo contengono spesso informazioni sul flusso, ad esempio "Se il cliente seleziona **Inoltra**, viene visualizzata la pagina Ordine," possono fornire informazioni utili per la creazione di grafici relativi al flusso di acquisti.

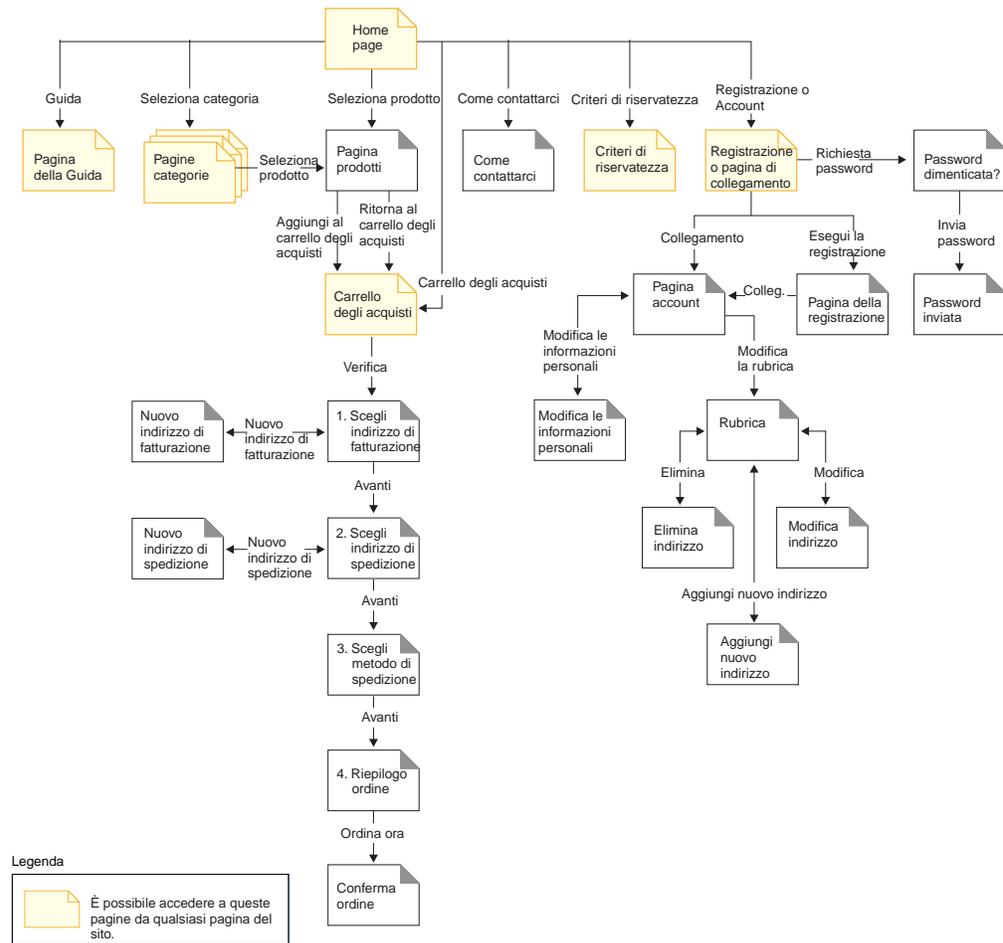
Il flusso di acquisti riflette i requisiti e i processi aziendali definiti per il negozio e descrive il modo in cui un cliente si sposta nel negozio stesso. Ad esempio,   possibile fare in modo che i clienti accedano al sito dalla home page, eseguendo poi la registrazione se desiderano sfogliare il catalogo, oppure consentire ai clienti di visualizzare il catalogo come ospiti, senza eseguire la registrazione. Alcuni flussi di acquisti consentono ai clienti di completare una "verifica rapida", mentre altri richiedono che il cliente completi tutti i passi della verifica ogni volta che fa un acquisto. In alternativa, il flusso di acquisti pu  offrire ai clienti la possibilit  di scegliere tra le due verifiche.



Per verificare che il grafico del flusso di acquisti sia completo, accertarsi che tutti i passi dei casi di utilizzo relativi al negozio siano illustrati nel grafico del flusso del negozio.

Se si segue il flusso di acquisti, come per il grafico di seguito riportato per il flusso di acquisti del negozio di esempio InFashion,   possibile vedere come i clienti si

spostano nel negozio.



Il grafico per il flusso di acquisti InFashion è piuttosto semplice. Tale grafico include il flusso principale del passaggio di un cliente attraverso il negozio ma non include nessuno scenario di errore. Ad esempio, cosa accade quando un cliente effettua la registrazione utilizzando una password errata oppure immette un numero di carta di credito non valido? Tuttavia, anche un grafico semplice come questo consente di sviluppare un elenco di pagine necessarie per il negozio. Per iniziare, è necessario creare una visualizzazione per ciascuna pagina del grafico del flusso di acquisti.

Ad esempio, se si desidera creare un negozio con lo stesso flusso di acquisti del grafico InFashion, è necessario creare le seguenti pagine:

Nota: Nella tabella di seguito riportata vengono elencati i nomi delle viste utilizzati per il negozio InFashion

Pagine del grafico del flusso di acquisti di InFashion (come visto dal cliente)	Visualizzazione corrispondente
Home page	StoreCatalogDisplayView
Pagina della guida	HelpView
Come contattarci	ContactView
Criteri di riservatezza	PrivacyView
Pagina di registrazione o di collegamento	LogonForm

Pagine del grafico del flusso di acquisti di InFashion (come visto dal cliente)	Visualizzazione corrispondente
Password dimenticata?	LogoffView
Password inviata	ResetPasswordForm
Pagina Account	LogonForm
Modifica le informazioni personali	UserRegistrationForm
Rubrica	AddressBookForm
Aggiungi nuovo indirizzo	AddressForm
Cancella indirizzo	AddressBookForm
Modifica indirizzo	AddressForm
Pagina Registrazione	UserRegistrationForm
Carrello degli acquisti	OrderItemDisplayViewShiptoAssoc
Scegli indirizzo di fatturazione	OrderItemDisplayViewShiptoAssoc
Nuovo indirizzo di fatturazione	OrderItemDisplayViewShiptoAssoc
Scegli indirizzo di spedizione	OrderItemDisplayViewShiptoAssoc
Nuovo indirizzo di spedizione	AddressForm
Scegli metodo di spedizione	OrderItemDisplayViewShiptoDsp
Riepilogo ordine	OrderDisplayPendingView
Conferma ordine	OrderOKView

Nota: Molte viste utilizzate in InFashion sono state create in modo specifico per InFashion. Tali viste sono elencate nel file `command.xml` dell'archivio negozio InFashion. Per ulteriori informazioni, fare riferimento alla sezione "Registrazione di comandi, viste e URL in WebSphere Commerce" a pagina 51.

La tabella sopra riportata presenta solo l'insieme base delle pagine da creare. Per stabilire le altre pagine da creare, è possibile osservare più attentamente i casi di utilizzo o gli altri metodi utilizzati per definire i processi aziendali.

Pagine di errore: I flussi di eccezione nei casi di utilizzo possono anche aiutare a stabilire le pagine di errore necessarie per la creazione del negozio. Il caso di utilizzo della registrazione per InFashion specifica i seguenti flussi di eccezione:

- L'indirizzo e-mail risulta già esistente: Se l'indirizzo e-mail risulta già presente nel sistema, verrà visualizzato un messaggio di errore, che richiede all'utente di fornire un altro indirizzo. La procedura ricomincia dall'inizio.
- Campi obbligatori mancanti: Se uno dei seguenti campi (E-mail, Password, Verifica password, Nome, Cognome) è vuoto, il sistema riporta un messaggio di errore. La procedura ricomincia dall'inizio.
- Password non valida: Se la password non corrisponde alla password di verifica, il sistema invia un messaggio di avvertenza.

Come risultato, sarà necessario creare una pagina di errore o un messaggio di errore per ciascun flusso di eccezione.

Sviluppo di un elenco di URL di comandi e viste

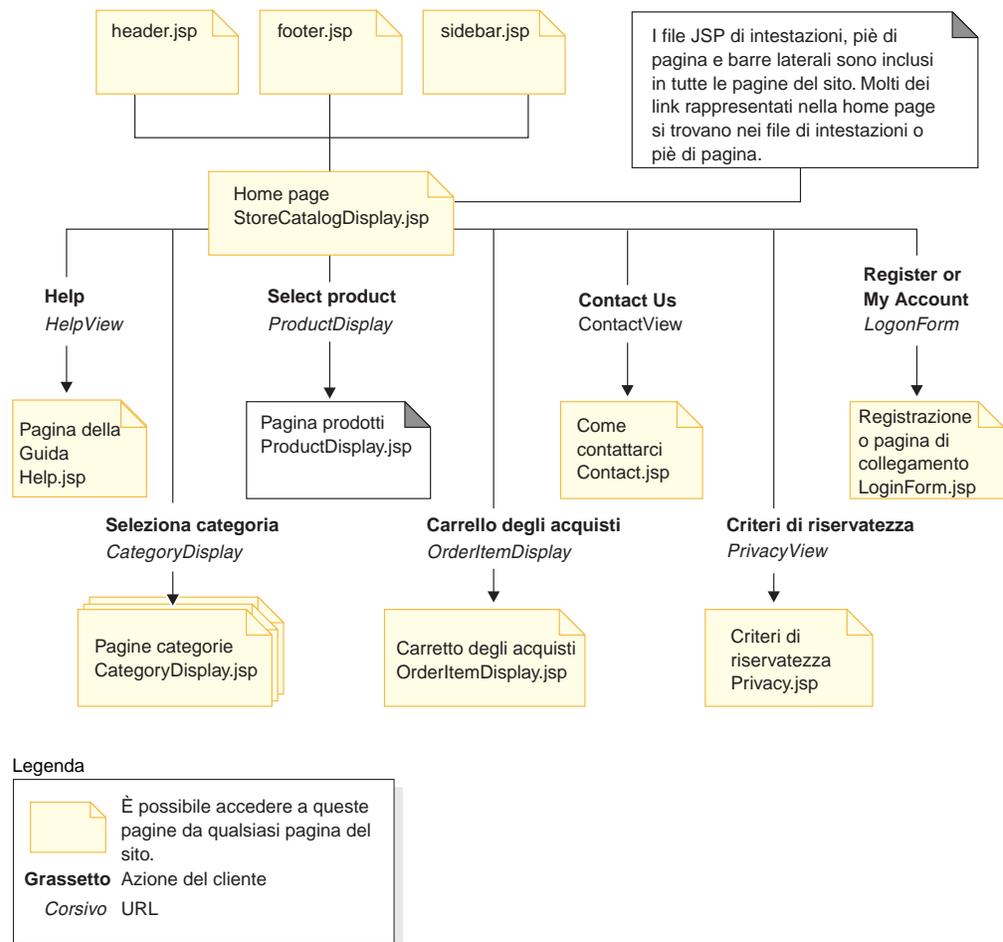
Come dimostrato nel grafico del flusso di acquisti di InFashion, i processi aziendali quali la verifica e la registrazione possono richiedere diverse pagine. Per unire

queste pagine in un processo aziendale o flusso attivo anziché in una semplice raccolta di pagine, è necessario includere nelle pagine comandi e viste.

Sviluppo di un elenco di URL necessari

Dopo aver sviluppato un elenco di pagine necessarie per la creazione del negozio, è necessario sviluppare un elenco di URL di comandi e viste necessari a realizzare i processi aziendali relativi al negozio. Utilizzando il grafico del flusso di acquisti del negozio e l'elenco di comandi e viste predefiniti, è possibile identificare gli URL necessari al completamento di ciascuna azione.

Gli URL di comandi e viste utilizzati nei negozi di esempio consentono di identificare gli URL necessari nel negozio. L'illustrazione di seguito riportata identifica gli URL relativi ad alcune azioni nel grafico del flusso di acquisti di InFashion. Per maggiori dettagli, fare riferimento alle informazioni sui negozi di esempio nella guida in linea di WebSphere Commerce.



Associazione di nomi file JSP e viste

WebSphere Commerce Server utilizza i comandi di visualizzazione per comporre una visualizzazione in risposta ad una richiesta. WebSphere Commerce Server fornisce i seguenti comandi di visualizzazione:

- `HttpForwardViewCommandImpl`: Questo comando inoltra la richiesta di visualizzazione ad un file JSP.
- `HttpRedirectViewCommandImpl`: Questo comando reindirizza la richiesta di visualizzazione ad un altro URL.

- `HttpDirectViewControllerImpl`: Questo tipo di comando invia la visualizzazione della risposta direttamente al client. Non richiama un file JSP. Le viste dirette consentono ai comandi di controllo di produrre la risposta di emissione (anziché il comando di visualizzazione).

Utilizzare il comando di visualizzazione `HttpForwardViewControllerImpl` per restituire i file JSP direttamente. Ad esempio, nel grafico che illustra gli URL utilizzati in *InFashion*, per visualizzare la pagina della guida (`Help.jsp`), `HelpView` è stato registrato nel registro delle visualizzazioni e associato a `Help.jsp` e al comando `HttpForwardViewControllerImpl`. Ciò è dimostrato nel seguente esempio:

```
<viewreg
viewname="HelpView"
devicefmt_id="-1"
storeent_id="@storeent_id_1"
interfacename="com.ibm.commerce.command.ForwardViewCommand"
classname="com.ibm.commerce.command.HttpForwardViewControllerImpl"
properties="docname=help.jsp"
internal="0"
https="0"
/>
```

Per l'interfaccia e la classe di implementazione viene utilizzato il nome classe completo.

Nota: In questo esempio, l'URL che richiama la visualizzazione non è limitato. Ciò significa che chiunque può accedere all'URL direttamente. Se si utilizza questa tecnica, verificare che il file JSP restituisca solo dati pubblici.

Utilizzare il comando `HttpForwardViewControllerImpl` per rappresentare le visualizzazioni restituite da un comando di visualizzazione. Un comando di visualizzazione legge i dati dal database ma non li modifica. Ad esempio, nel grafico che illustra gli URL utilizzati in *InFashion*, il comando `OrderItemDisplay` restituisce la visualizzazione `OrderItemDisplayViewShiptoAssoc`. Quando questa visualizzazione è stata registrata nel relativo registro, sono stati associati il file `OrderItemDisplay.jsp` e il comando `HttpForwardViewControllerImpl`. Ciò è dimostrato nel seguente esempio:

```
<viewreg
viewname="OrderItemDisplayViewShiptoAssoc"
devicefmt_id="-1"
storeent_id="@storeent_id_1"
interfacename="com.ibm.commerce.command.ForwardViewCommand"
classname="com.ibm.commerce.command.HttpForwardViewControllerImpl"
properties="docname=OrderItemDisplay.jsp"
internal="0"
https="0"
/>
```

Nota: In questo esempio, l'URL che richiama la visualizzazione è limitato. Ciò significa che solo gli utenti con accesso designato possono richiamare l'URL direttamente. L'accesso a questa visualizzazione è controllato da chiunque abbia accesso all'URL relativo al comando.

E' necessario assegnare un nome file JSP a ciascuna visualizzazione associata ad un comando di visualizzazione (ad esempio, `OrderItemDisplay`) utilizzato. Per ulteriori informazioni sull'associazione dei nomi file JSP con le visualizzazioni, fare riferimento alla sezione "Registrazione di comandi, viste e URL in *WebSphere Commerce*" a pagina 51.

Nota: ProductDisplay e CategoryDisplay visualizzano i nomi file JSP associati nei dati del catalogo anziché nel registro di visualizzazione.

Utilizzare il comando `HttpRedirectViewCommandImpl` per rappresentare le visualizzazioni restituite da un comando che modifica il database. Per utilizzare la visualizzazione `redirect`, specificare il nome della visualizzazione utilizzando il parametro `&URL=` dell'URL. Ad esempio, quando si aggiungono informazioni sull'indirizzo in `AddressForm` del negozio di esempio `InFashion` e si fa clic su **Inoltra**, viene richiamato il comando `AddressAdd`. L'URL utilizzato per richiamare il comando `AddressAdd` specifica `AddressBookForm` come visualizzazione mediante il parametro `&URL=`. Come risultato, si ritorna alla visualizzazione `AddressBookForm`. Quando la visualizzazione `AddressBookForm` è stata registrata nel relativo registro, sono stati associati il file `AddressBookForm.jsp` e il comando `HttpForwardViewCommandImpl`.

E' necessario utilizzare la tecnica `URL=parameter` per tutti i comandi di non visualizzazione. I comandi di non visualizzazione sono comandi che provocano modifiche ai dati del database.

Parte 3. Panoramica sui dati del negozio

Capitolo 4. Dati del negozio

In questo capitolo viene fornita una panoramica sull'architettura dei dati del negozio di WebSphere Commerce Server e sulle risorse dati che creano un negozio. In questo capitolo è inserito anche il modello di informazioni di WebSphere Commerce Server.

Cosa sono i dati del negozio

I dati del negozio sono le informazioni caricate nel database di WebSphere Commerce Server, che consentono a un negozio di funzionare. Per funzionare correttamente, un negozio deve avere i dati necessari per supportare tutte le attività dei clienti. Ad esempio, perché un cliente possa fare un acquisto, il negozio deve disporre di un catalogo delle merci in vendita (dati del catalogo), di dati associati all'elaborazione degli ordini (tasse e dati sulla spedizione) e dell'inventario per la compilazione della richiesta (dati sull'inventario e la compilazione).

Modello di informazioni dati del negozio

Questa guida utilizza un modello di informazioni per illustrare il modo in cui i dati del negozio sono strutturati in WebSphere Commerce Server. Il modello di informazioni di WebSphere Commerce Server è un estratto di grande qualità delle informazioni contenute nei modelli di dati e oggetti di WebSphere Commerce Server. Il modello di informazioni evidenzia le funzioni più importanti dei modelli di dati e oggetti, ma non include i dettagli minimi specifici per la realizzazione dello schema e dell'oggetto.

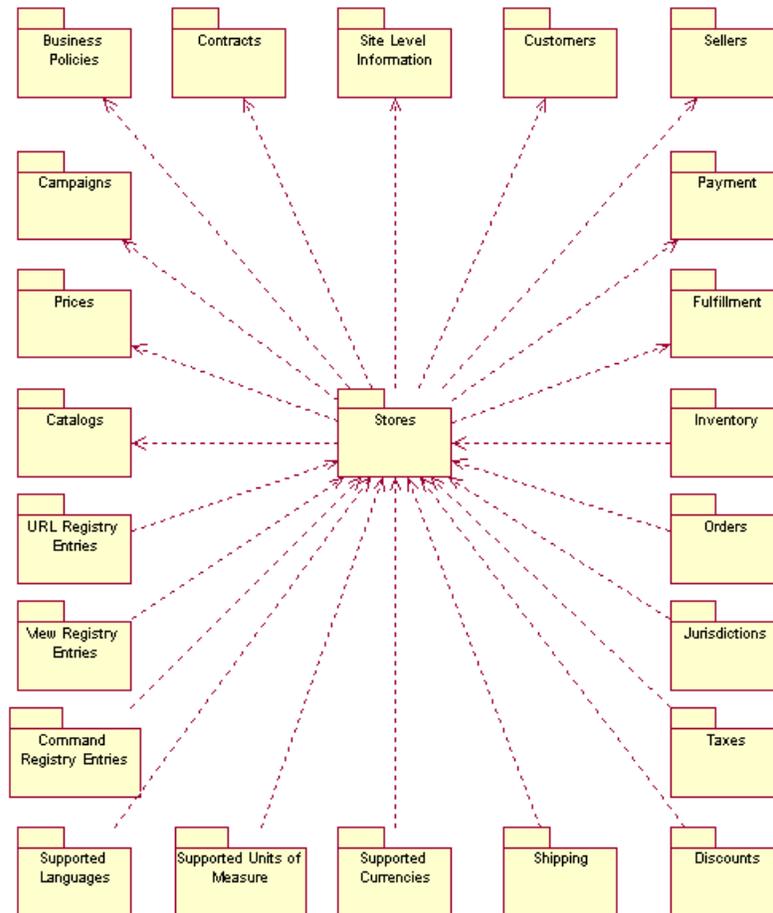
Ad esempio, alcune tabelle e oggetti nei modelli di dati o oggetti che contengono dati relativi alle relazioni tra entità (ad esempio coppie di chiavi esterne) non vengono visualizzati come entità nel modello di informazioni. Al contrario, queste relazioni tra entità sono implicite nelle righe delle relazioni tra entità nei modelli di informazioni. Il modello di informazioni, inoltre, non illustra le estensioni *dettagliate* (attributi di dati supplementari di un'entità memorizzata in un'altra tabella in seguito a un'implementazione: ad esempio, la descrizione di un prodotto è un'estensione memorizzata separatamente dell'entità del prodotto). Per ulteriori informazioni sugli oggetti di relazione e le estensioni dettagliate, fare riferimento al modello di oggetti nella guida in linea di WebSphere Commerce.



Per ulteriori informazioni sui modelli di oggetti e di dati di WebSphere Commerce, fare riferimento alla guida in linea di WebSphere Commerce.

Risorse dati del negozio

Il grafico di seguito riportato illustra le risorse dati di un negozio di WebSphere Commerce.



Questo grafico, insieme a tutti gli altri grafici della sezione relativa ai dati del negozio fanno parte del modello di informazioni di WebSphere Commerce. Per ulteriori informazioni sul modello di informazioni, fare riferimento alla sezione "Modello di informazioni dati del negozio" a pagina 27. Per ulteriori informazioni sulle convenzioni utilizzate in questo grafico, fare riferimento all'Appendice A, "Legenda UML" a pagina 317.

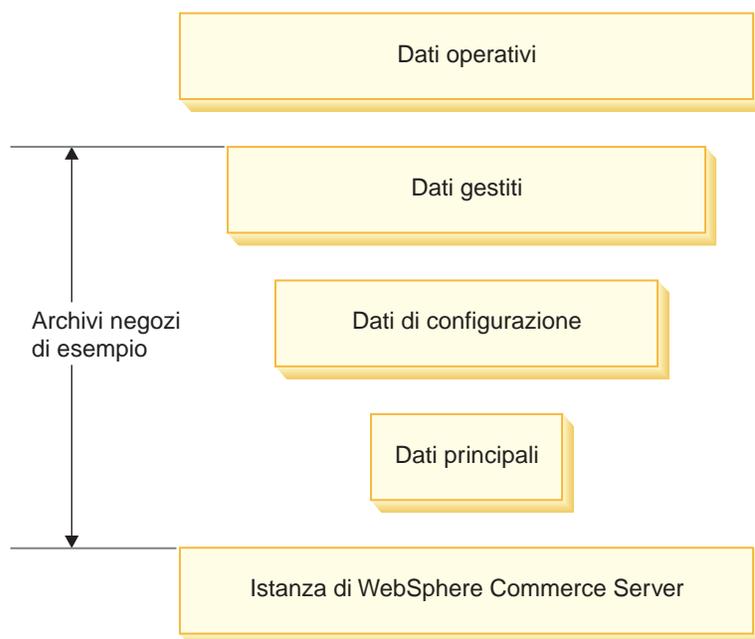
Le frecce del grafico seguono due direzioni. In alcuni casi, la freccia punta verso il negozio, ad esempio da Valuta a Negozi. In questo caso, le risorse valuta sono esclusive per questo particolare negozio e rappresentano parte del negozio. Si tratta delle valute supportate da questo negozio. Se il negozio viene cancellato, deve essere cancellato anche l'elenco delle valute supportate dal negozio.

Quando una freccia punta da Negozi ad una risorsa, ad esempio Catalogo, la risorsa può essere condivisa da altri negozi. Un catalogo può essere condiviso da vari negozi. Tuttavia, se il negozio viene cancellato, il catalogo continua ad esistere.

Ciascuna risorsa dati illustrata nel grafico sopra riportato viene esaminata in maggiori dettagli nelle sezioni della Parte 4, "Sviluppo dei dati del negozio" a pagina 37

Architettura dati del negozio

I dati presenti nei negozi di WebSphere Commerce sono conformi all'architettura illustrata nel grafico di seguito riportato. Ciascuna delle risorse dati del negozio illustrata nel grafico della sezione "Risorse dati del negozio" a pagina 28, può essere classificata come appartenente a uno o più tipi di dati del negozio illustrati sotto.



Istanza del server di WebSphere Commerce

Il livello base di dati è contenuto nell'istanza di WebSphere Commerce Server. Quando viene creata un'istanza, i file bootstrap, caricati in formato XML, inseriscono le informazioni nel database. I file bootstrap creano i seguenti tipi di dati:

- Tipi di calcolo, tipi di unità (browser, e-mail, I-Mode e così via), tipi di messaggio e ruoli
- ID di gestione predefinito, WCSADMIN
- Comandi, viste e URL predefiniti
- Politiche aziendali predefinite
- Lingue e valute supportate dall'istanza
- Organizzazione predefinita, che può essere utilizzata come proprietario del negozio
- Organizzazione sito predefinita
- Gruppo negozi predefinito

Queste informazioni sono disponibili per tutti i negozi presenti in questa istanza e vengono identificate come informazioni a livello di sito nel grafico della sezione "Risorse dati del negozio" a pagina 28.

Per informazioni più dettagliate sui file bootstrap e sulle tabelle di database da essi costituite, fare riferimento alla guida in linea di WebSphere Commerce.

Dati principali

Il livello successivo ai dati del negozio è costituito dai dati principali. I dati principali rappresentano i dati minimi per un negozio e includono:

- L'ID del negozio nella tabella STOREENT. L'ID crea un negozio nel database.
- Il contratto predefinito.
- L'ID del negozio nelle tabelle di database del contratto.
- L'ID del membro per l'organizzazione che gestisce il negozio nelle tabelle di database del contratto.
- La directory del negozio nella tabella STORE. La directory del negozio è la directory in cui si trovano le risorse Web del negozio.
- Il soprannome o l'identificativo per l'indirizzo del negozio nella tabella STADDRESS. Il soprannome è univoco per ciascun negozio.

Se per creare un negozio si utilizza Servizi del negozio, queste informazioni vengono create con un nuovo archivio negozio. Servizi del negozio consente di selezionare l'organizzazione predefinita che può agire come proprietario del negozio oppure è possibile creare un'altra organizzazione che agisca come proprietario utilizzando la Console di gestione. Se per creare un negozio non viene utilizzato Servizi del negozio, è necessario caricare queste informazioni nel database mediante il Pacchetto Loader oppure modificare il database direttamente.

I dati Negozi nel grafico della sezione "Risorse dati del negozio" a pagina 28 sono dati principali.

Dati di configurazione

I dati di configurazione controllano il runtime di commerce server. Il runtime del server comune fornisce la struttura in cui vengono distribuite ed eseguite le applicazioni commerciali. La struttura si compone del modello di programmazione, il modello di elaborazione, la gestione delle eccezioni, il controllo delle transazioni, l'accesso ai dati e il modello di conservazione. Il runtime del server comune agisce sui servizi di runtime forniti da WebSphere Application Server per supportare le applicazioni di WebSphere Commerce Server. I dati di configurazione identificano i comandi, le viste e i file JSP che il negozio userà per visualizzare le pagine.

Le risorse dati di seguito riportate, identificate nel grafico della sezione "Risorse dati del negozio" a pagina 28 sono classificate come dati di configurazione:

- Voci del registro dei comandi
- Voci del registro delle visualizzazioni
- Voci del registro degli URL

Dati gestiti

I dati gestiti sono dati creati dal venditore, di sola lettura per i clienti del sito del venditore. Poiché il venditore ne ha il completo controllo dello stato, questi dati possono essere gestiti mediante un sistema di gestione del contenuto.

Le risorse dati di seguito riportate, identificate nel grafico della sezione "Risorse dati del negozio" a pagina 28, sono classificate come dati gestiti:

- Campagne
- Politiche aziendali
- Contratti
- Centri di evasione ordini
- Giurisdizioni
- Tasse

- Sconti
- Spedizione
- Valute
- Unità di misura
- Lingue
- Cataloghi
- Prezzi
- Clienti
- Venditori
- Pagamento

Dati operativi

I dati operativi sono i dati creati o modificati (direttamente o indirettamente) dai clienti del sito in seguito alle loro interazioni con il sito. Ad esempio, gli ordini effettuati dai clienti sono considerati dati operativi, in quanto rappresentano livelli di inventario che salgono e scendono in base al funzionamento del negozio. Anche i clienti sono considerati dati operativi. I dati creati dal venditore possono essere operativi.

Poiché le modifiche ai dati operativi non sono sotto il completo controllo del venditore, non ha senso gestire questi dati con un sistema di gestione del contenuto.

Le risorse dati di seguito riportate, identificate nel grafico della sezione “Risorse dati del negozio” a pagina 28 sono classificate come dati operativi:

- Ordini
- Inventario
- Evasione ordini
- Clienti

Nota: In alcuni casi, la linea tra dati operativi e dati gestiti può essere difficile da stabilire. Ad esempio, se in un negozio i dati relativi ai clienti e al contratto possono essere considerati dati gestiti, in un altro negozio, lo stesso tipo di dati può essere considerato operativo. Il primo negozio può gestire i dati dei clienti e i relativi contratti, in quanto controlla un insieme specifico di clienti, vale a dire i clienti che non possono effettuare la registrazione in linea. Il secondo negozio, invece, consente ai clienti la registrazione in linea e crea informazioni sui contratti in linea.

Un secondo esempio riguarda i dati dei cataloghi. Nel sito di un unico venditore, il catalogo è considerato dato gestito. Nel sito di un mercato, i dati del catalogo possono essere considerati operativi.

In alcuni siti, alcuni record dello stesso tipo di dati possono essere considerati gestiti mentre altri record sono considerati operativi. Ad esempio, il contratto predefinito può essere considerato un dato gestito, ma gli specifici contratti negoziati in linea sono dati operativi.

Architettura dati del negozio e negozi di esempio

I negozi di esempio forniti con WebSphere Commerce includono molti tipi di dati presenti nell'architettura dei dati del negozio. Ad esempio, è necessario che esista un'istanza di WebSphere Commerce Server perché un negozio possa essere creato

mediante un negozio di esempio o un negozio di esempio possa essere pubblicato. Quindi, se si crea un negozio sulla base di un negozio di esempio utilizzando gli strumenti Servizi del negozio, vengono creati i dati principali. I negozi di esempio presentano tutta la configurazione necessaria e molti dei dati gestiti necessari per il funzionamento di un negozio. Quando si creano i negozi in base a determinati negozi di esempio, è possibile visualizzare le istruzioni per il completamento di alcune impostazioni di dati, utilizzando gli strumenti di WebSphere Commerce Accelerator.

Strumenti per la creazione di dati

WebSphere Commerce fornisce vari strumenti per la creazione e la manipolazione dei dati del negozio. Tali strumenti sono di seguito elencati:

Pacchetto WebSphere Commerce Loader

Il Pacchetto Loader è composto principalmente da programmi di utilità per la preparazione e il caricamento dei dati in un database di WebSphere Commerce. Per ulteriori informazioni, consultare Parte 7, “Pubblicazione del negozio” a pagina 213.

Servizi del negozio

Servizi del negozio modifica i dati precedentemente pubblicati sotto forma di un archivio negozio e non i dati presenti nel database. Servizi del negozio, inoltre, consente di pubblicare tutte le risorse dati del negozio presenti nel database. Per ulteriori informazioni, consultare Parte 7, “Pubblicazione del negozio” a pagina 213.

Console di gestione

La Console di gestione consente di controllare il sito o il negozio completando operazioni di amministrazione ed attività di configurazione. E' inoltre possibile utilizzare la Console di gestione per creare nuove organizzazioni ed utenti e per assegnare utenti a ruoli (Sviluppatore del negozio, Responsabile del negozio, Responsabile del sito e così via). La Console di gestione consente inoltre di identificare i tipi di notifiche e messaggi disponibili nel negozio.

WebSphere Commerce Accelerator

WebSphere Commerce Accelerator è un'area di lavoro di strumenti in linea che consente di creare e mantenere varie risorse del negozio. Gran parte dei dati del negozio possono essere creati e gestiti mediante gli strumenti di WebSphere Commerce Accelerator. Tuttavia, in alcuni casi, determinati dati devono essere già stati caricati nel database, mediante gli strumenti di pubblicazione di Servizi del negozio o del Pacchetto Loader, perché sia possibile creare altri dati mediante WebSphere Commerce Accelerator. Per ulteriori informazioni, fare riferimento al “Grafico di riepilogo degli strumenti e dei dati del negozio”.

Console di gestione dell'organizzazione

La Console di gestione dell'organizzazione consente la gestione delle organizzazioni che accedono al sito o negozio. La Console di gestione dell'organizzazione consente inoltre al responsabile acquirenti di gestire gli acquirenti della propria organizzazione.

Grafico di riepilogo degli strumenti e dei dati del negozio

Il grafico di seguito riportato illustra gli strumenti che è possibile utilizzare per creare ciascun tipo di dati.

Strumenti per la creazione di dati	Dati principali	Dati di configurazione	Dati gestiti	Dati operativi
Pacchetto WebSphere Commerce Loader	Utilizzare il pacchetto Loader per caricare i dati principali sotto forma di un file XML. Per ulteriori informazioni, consultare la sezione "Creazione di risorse dati del negozio in un file XML" a pagina 45.	Utilizzare il pacchetto Loader per caricare i dati di configurazione sotto forma di un file XML. Per ulteriori informazioni, consultare la sezione "Creazione di un file XML per registrare comandi, viste e URL" a pagina 51.	Utilizzare il pacchetto Loader per caricare i dati gestiti sotto forma di un file XML. Per ulteriori informazioni, fare riferimento ai capitoli corrispondenti sulle risorse dati gestiti.	In generale, i dati operativi non possono essere caricati con il pacchetto Loader.
Servizi del negozio (per i dati precedentemente pubblicati sotto forma di un archivio negozio)	Quando si crea un nuovo archivio negozio utilizzando Servizi del negozio, vengono creati i dati principali. Per ulteriori informazioni sull'uso di Servizi del negozio, fare riferimento alla guida in linea di WebSphere Commerce.	Non applicabile.	Servizi del negozio consente di creare e modificare parti delle seguenti risorse gestite: <ul style="list-style-type: none"> • Giurisdizioni • Tasse • Spedizione • Valuta • Lingue Per ulteriori informazioni sulle parti delle risorse database che Servizi del negozio consente di modificare o creare, fare riferimento alla sezione della guida in linea di WebSphere Commerce "Modifica delle risorse database del negozio".	Non applicabile.
Console di gestione	Utilizzare la Console di gestione per creare un'organizzazione che agisca come proprietario del negozio.	Non applicabile.	Non applicabile.	Non applicabile.

Strumenti per la creazione di dati	Dati principali	Dati di configurazione	Dati gestiti	Dati operativi
WebSphere Commerce Accelerator	Non applicabile.	Non applicabile.	<p>Utilizzare WebSphere Commerce Accelerator per creare o modificare i dati di seguito riportati:</p> <ul style="list-style-type: none"> • Campagne • Contratti (un contratto predefinito deve essere presente nel database perché gli strumenti Business Relationship Management di WebSphere Commerce Accelerator possano essere utilizzati per creare altri contratti o modificare i contratti esistenti. Utilizzare il Pacchetto Loader o Servizi del negozio per creare un contratto predefinito nel database). 	<p>I clienti creano dati operativi quando eseguono la registrazione al negozio o fanno shopping. In alcuni casi, tuttavia, è possibile utilizzare WebSphere Commerce Accelerator per inserire gli ordini di un cliente o per creare una restituzione.</p> <p>WebSphere Commerce Accelerator consente anche di gestire l'inventario.</p>

Strumenti per la creazione di dati	Dati principali	Dati di configurazione	Dati gestiti	Dati operativi
WebSphere Commerce Accelerator (continua)	Non applicabile.	Non applicabile.	<ul style="list-style-type: none"> • Gestione • Sconti • Cataloghi (un catalogo principale deve essere presente nel database perché gli strumenti Product Management di WebSphere Commerce Accelerator possano essere utilizzati per creare un catalogo di esplorazione e aggiungere o modificare le informazioni sul prodotto. Utilizzare il Pacchetto Loader o Servizi del negozio per creare un catalogo principale nel database). • Prezzi 	Non applicabile.
Console di gestione della organizzazione	Non applicabile.	Non applicabile.	Non applicabile.	Clienti e acquirenti vengono creati quando accedono al negozio. Tuttavia, mediante la Console di gestione della organizzazione, è possibile approvare gli acquirenti o crearne di nuovi.

Parte 4. Sviluppo dei dati del negozio

I capitoli presenti in questa sezione illustrano in maggiori dettagli ciascuna delle risorse dati del negozio. Le risorse dati del negozio presenti in questa sezione sono organizzate in base alla struttura dell'architettura dei dati del negozio di WebSphere Commerce:

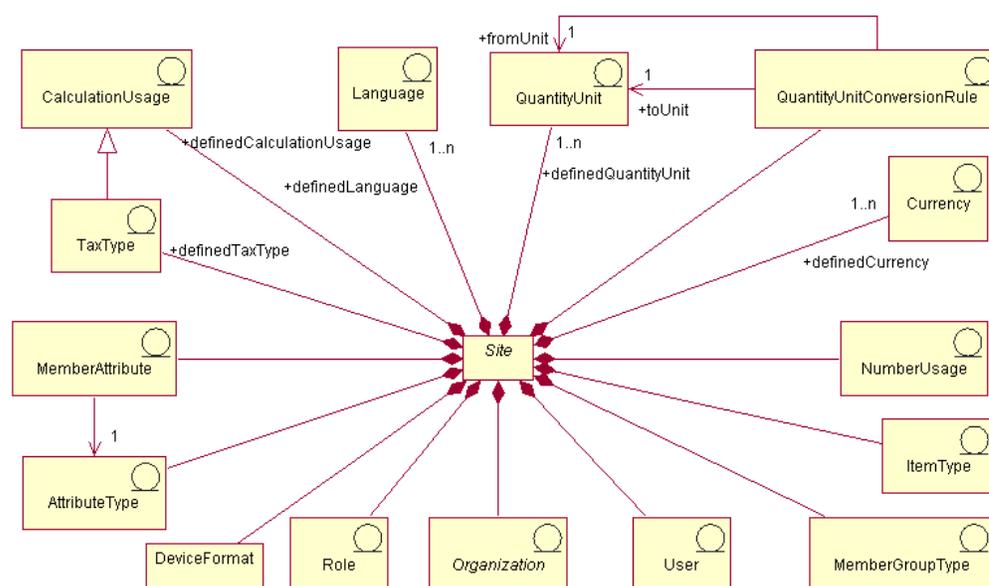
- Istanza del server di WebSphere Commerce
 - Sito
- Dati principali
 - Negozio
- Dati di configurazione
 - Registro comandi
 - Registro visualizzazioni
 - Registro URL
- Dati gestiti
 - Risorse condivise
 - Catalogo
 - Prezzi
 - Contratti (incluse le politiche aziendali)
 - Evasione ordini
 - Campagne
 - Pagamento
 - Limitati alle risorse del negozio
 - Lingue supportate
 - Valute supportate
 - Unità di misura supportate
 - Giurisdizioni
 - Spedizione
 - Tassazione
 - Sconti
- Dati operativi
 - Inventario
 - Ordini
 - Clienti

Capitolo 5. Risorse del sito

Ciascuna istanza di WebSphere Commerce Server presenta un database di informazioni relazionali. Un'istanza viene creata con i file bootstrap utilizzati per inserire le informazioni nelle tabelle di database, dopo la creazione dello schema. Una volta caricati i dati, è possibile consultare le informazioni di precaricamento nelle tabelle di database appropriate. Molte tabelle di database contengono informazioni a livello di negozio o di gruppo di negozi. Queste informazioni sono generalmente gestite dai responsabili del negozio. Tuttavia, alcune tabelle contengono informazioni che rappresentano le capacità a livello di sito di WebSphere Commerce, disponibili per tutti i negozi dell'istanza e sono gestite dal responsabile del sito di WebSphere Commerce. Queste capacità sono esaminate in questo capitolo. Per ulteriori informazioni sui file bootstrap, fare riferimento alla guida in linea di WebSphere Commerce. Per ulteriori informazioni, consultare la sezione Capitolo 6, "Risorse del negozio" a pagina 43.

Risorse del sito in WebSphere Commerce

Il grafico di seguito riportato illustra i tipi di dati del sito e le relazioni con il sito.



Per ulteriori informazioni sulle convenzioni utilizzate in questo grafico, fare riferimento all'Appendice A, "Legenda UML" a pagina 317. Questo grafico, insieme a tutti gli altri grafici della sezione relativa ai dati del negozio fa parte del modello di informazioni di WebSphere Commerce. Per ulteriori informazioni sul modello di informazioni, fare riferimento alla sezione "Modello di informazioni dati del negozio" a pagina 27.

Lingua

Un sito può definire molte *lingue* nella tabella LANGUAGE e descrivere le lingue nella tabella LANGUAGEDS. Ogni negozio può supportare generalmente un sottoinsieme di tali lingue, aggiungendo righe alla tabella STORELANG. Le dieci

lingue predefinite sono: tedesco, cinese tradizionale e semplificato, giapponese, coreano, italiano, francese, spagnolo, portoghese brasiliano e inglese.

Attributi membro

Gli *attributi membro* si trovano nella tabella MBRATTR e rappresentano l'insieme di nomi di attributi definiti per i quali possono esistere dei valori relativi a organizzazioni o utenti. Esempi di tali nomi di attributi comprendono JobFunction, ProcurementCard, SpendingLimit, ReferredBy e CountryOfOperation. I valori di attributi per determinate organizzazioni o utenti possono trovarsi nella tabella MBRATTRVAL e possono essere diversi per i diversi negozi o gruppi di negozi.

Tipi di attributi

I *tipi di attributi* si trovano nella tabella ATTRTYPE e rappresentano i tipi di dati definiti che possono essere utilizzati per rappresentare i valori di attributi. Esempi di tipi di dati includono INTEGER, STRING e FLOAT.

Tipi di gruppi di membri

I *tipi di gruppi di membri* si trovano nella tabella MBRGRPTYPE e rappresentano l'insieme di utilizzi di gruppi di membri definiti. Ai gruppi di membri vengono assegnati gli utilizzi, aggiungendo righe alla tabella MBRGRPUSG. Esempi di utilizzi di gruppi di membri includono AccessGroup (per l'utilizzo con le politiche di controllo accessi) e UserGroup (per scopi generali, ad esempio gruppi di clienti).

Utente

Con *utente* si indicano le identità di un utente autenticato. Gli utenti sono generalmente clienti che eseguono o approvano gli ordini per conto di organizzazioni di acquisti, agenti di vendita che elaborano gli ordini per organizzazioni di vendita o gestiscono le risorse a livello di negozio o responsabili del sito che gestiscono l'istanza di WebSphere Commerce Server. Ciascun utente è associato ad un sito ed è definito nella tabella USERS.

Organizzazione

Con *organizzazione* si indicano le organizzazioni e le unità organizzative all'interno delle organizzazioni. Le organizzazioni sono generalmente entità aziendali responsabili degli acquisti e delle vendite. Gli ordini effettuati dai clienti in un'organizzazione di acquisti business-to-business sono registrati come se fossero stati effettuati per conto dell'organizzazione di acquisti. I negozi, i cataloghi ed i centri di evasione ordini sono gestiti da organizzazioni responsabili di alcuni aspetti della vendita. Le organizzazioni sono definite nella tabella ORGENTITY.

Ruolo

Con *ruolo* si indica l'insieme di ruoli definiti che possono essere assegnati agli utenti all'interno delle organizzazioni. Ad esempio, a un utente può essere assegnato il ruolo di rappresentante del servizio clienti all'interno di un'organizzazione di vendita o può essere assegnato il ruolo di responsabile approvazioni acquirenti all'interno di un'organizzazione di acquisti. I nomi e le descrizioni dei ruoli predefiniti si trovano nella tabella ROLE. Per ulteriori informazioni sugli specifici ruoli, fare riferimento alla guida in linea di WebSphere Commerce.

Conversione unità di quantità

Ciascun sito presenta *conversioni di quantità*. Si tratta di operazioni di moltiplicazione o divisione che possono essere utilizzate per convertire unità di misura diverse. Queste regole si trovano nella tabella QTYCONVERT.

Unità di quantità

Le *unità di quantità* rappresentano l'insieme di unità di misura per il sito. Tali unità sono definite nella tabella QTYUNIT e descritte nella tabella QTYUNITDSC. Ciascun negozio può specificare il modo in cui le quantità di ciascuna unità di misura vengono arrotondate e formattate per la visualizzazione, in base all'utilizzo previsto, aggiungendo righe alla tabella QTYFORMAT.

Tipi di tasse

I *tipi di tasse* rappresentano i vari utilizzi di calcoli delle tasse. Le tasse sulla vendita e le tasse di spedizione rappresentano due diversi utilizzi di calcoli. I tipi di tasse sono definiti nella tabella TAXTYPE.

Utilizzi del calcolo

Gli *utilizzi del calcolo* rappresentano i diversi tipi di calcolo che possono essere eseguiti dal comando OrderPrepare. Gli utilizzi del calcolo sono definiti per sconti, spedizione, tasse sulla vendita, tasse di spedizione ed e-coupon. Gli utilizzi del calcolo sono definiti nella tabella CALUSAGE.

Valuta

Ciascun sito definisce un numero di *valute* nella tabella SETCURR e descrive tali valute nella tabella SETCURRDSC. Ogni negozio può supportare un sottoinsieme di valute, aggiungendo righe alla tabella CURLIST, una riga per ciascuna valuta supportata.

Utilizzo numero

L'*utilizzo numero* rappresenta l'utilizzo previsto per i numeri. I negozi possono specificare regole di arrotondamento e formattazione diverse per i numeri visualizzati, in base al modo in cui vengono utilizzati. Ad esempio, un negozio può arrotondare i prezzi delle unità a quattro cifre decimali specificando l'utilizzo "prezzo unità" ma arrotondare altri importi a due cifre decimali specificando l'utilizzo "predefinito". L'utilizzo numero è definito nella tabella NUMBRUSG e descritto nella tabella NUMBRUSGDS.

Tipi di articoli

I *tipi di articoli* rappresentano i diversi tipi di articoli di base. I due tipi di articoli di base in WebSphere Commerce sono kit dinamico e articolo normale. I tipi di articoli sono predefiniti nella tabella ITEMTYPE. Per ulteriori informazioni sugli articoli di base, consultare il Capitolo 21, "Risorse inventario" a pagina 173.

Formati unità

I *formati unità* sono presenti nella tabella DEVICEFMT e rappresentano i diversi formati unità utilizzati da un sito, quali browser, I_MODE, e-mail, MQXML e MQNC. Tutti i tipi di unità consentono agli utenti di interagire con il sito attraverso vari supporti.

Nota: Per alcune risorse del sito, quali Lingua, Valuta, Unità di quantità e Regola di conversione unità di quantità, il responsabile del sito può estendere le

capacità a livello di sito aggiungendo righe alle tabelle appropriate. Per le altre risorse, delle personalizzazioni possono essere necessarie per estendere le capacità a livello di sito rappresentate dalle risorse. Ad esempio, se un responsabile di sito aggiunge un nuovo utilizzo numero per visualizzare subtotali con un simbolo di valuta personalizzato, il programma che visualizza i subtotali deve essere personalizzato in modo da poter specificare il nuovo utilizzo numero subtotale quando gli importi dei subtotali vengono formattati per la visualizzazione.

Creazione di risorse del sito in WebSphere Commerce

Le risorse del sito vengono create quando si crea un'istanza in WebSphere Commerce Server. Per ulteriori informazioni sulla creazione di un'istanza di WebSphere Commerce Server, fare riferimento al manuale *IBM WebSphere Commerce - Guida all'installazione*, Capitolo 5 "Creazione o modifica di un'istanza."

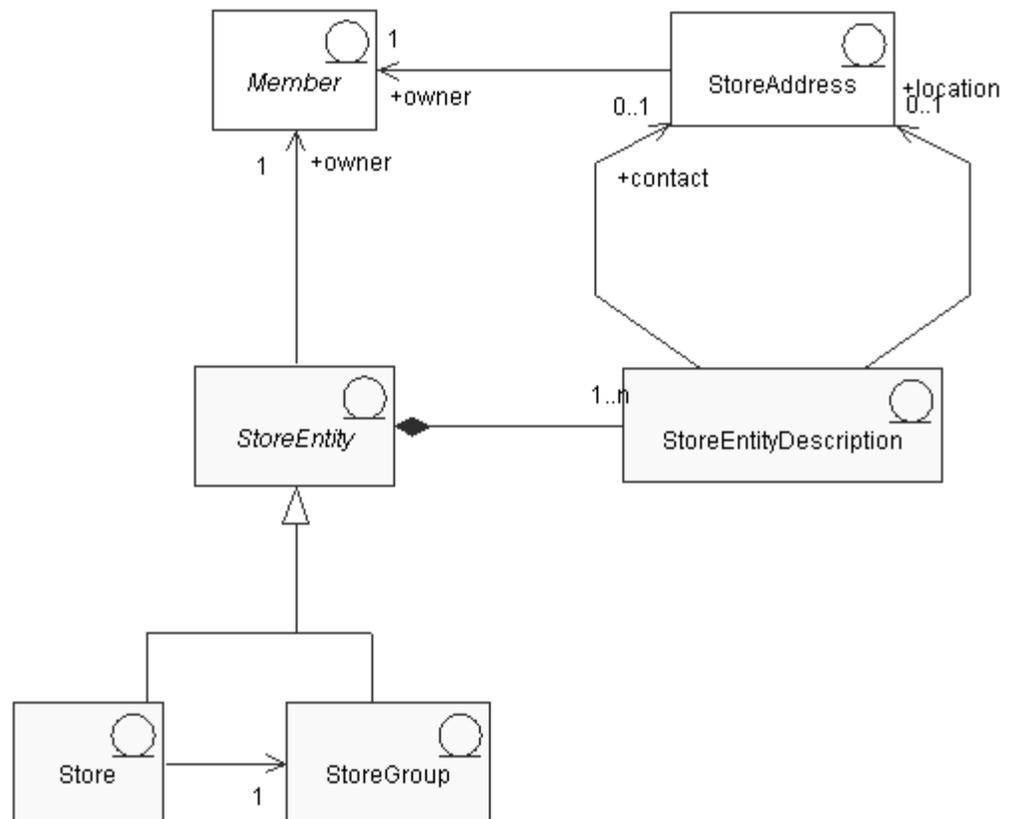
Capitolo 6. Risorse del negozio

Per creare un negozio in WebSphere Commerce, è necessario creare nel database quanto segue:

- Il negozio
- Il gruppo a cui il negozio appartiene
- L'oggetto entità negozio astratta che rappresenta allo stesso tempo un negozio o un gruppo di negozi

Risorse del negozio in WebSphere Commerce

Il grafico di seguito riportato illustra le risorse del negozio in WebSphere Commerce Server.



Questo grafico, insieme a tutti gli altri grafici della sezione relativa ai dati del negozio fa parte del modello di informazioni di WebSphere Commerce Server. Per ulteriori informazioni sul modello di informazioni, fare riferimento alla sezione "Modello di informazioni dati del negozio" a pagina 27. Per ulteriori informazioni sulle convenzioni utilizzate in questo grafico, fare riferimento all'Appendice A, "Legenda UML" a pagina 317.

Entità negozio

Un'entità *negozio* è una classe superiore astratta che può rappresentare sia un negozio che un gruppo di negozi.

Un'entità negozio ha un unico proprietario (un membro). Per ulteriori informazioni sui membri, fare riferimento al Capitolo 23, "Risorse clienti e rivenditori" a pagina 181.

Descrizione dell'entità negozio

Nella *descrizione dell'entità negozio* viene descritta l'entità negozio. Un'entità negozio può includere una descrizione. Se il negozio supporta più lingue, la descrizione dell'entità negozio può essere in più lingue. La descrizione può includere un indirizzo di contatto e un indirizzo di ubicazione per l'entità negozio.

Negozio

Un *negozio* è un'entità negozio. Un negozio deve far parte di un gruppo di negozi.

Gruppo di negozi

Un *gruppo di negozi* è una serie di negozi. Un gruppo di negozi è un'entità negozio. Il gruppo di negozi agisce come un contenitore per le informazioni comuni, che possono essere memorizzate a livello di gruppo di negozi e condivise da tutti i negozi del gruppo. Ad esempio, i negozi presenti nello stesso gruppo possono condividere informazioni quali categorie di tasse, lingue supportate, valute supportate, codici di calcolo e giurisdizioni di spedizione.

Al momento, può esistere un solo gruppo di negozi che può essere gestito a livello di amministrazione del sito all'interno di WebSphere Commerce Server.



Per informazioni più dettagliate sulla struttura delle risorse del negozio di WebSphere Commerce Server, fare riferimento ai modelli di oggetti e dati del negozio nella guida in linea di WebSphere Commerce.

Creazione di risorse del negozio in WebSphere Commerce

Lo strumento Servizi del negozio in WebSphere Commerce consente di creare o modificare le seguenti risorse del negozio:

- L'ID del negozio e l'ID del membro nelle risorse dei contatti
- L'ID del negozio nella tabella STOREENT
- La directory del negozio nella tabella STORE
- Il soprannome dell'indirizzo nella tabella STADDRESS
- La descrizione del negozio
- L'indirizzo del negozio

Nota: Gli strumenti Servizi del negozio utilizzano file XML precostituiti sotto forma di un archivio negozio.

Esistono due modi per creare le risorse del negozio:

- Modificare le risorse del negozio esistenti di uno dei negozi di esempio forniti con WebSphere Commerce o di un archivio negozio esistente.
- Creare le risorse del negozio sotto forma di file XML, che può essere pubblicato come parte di un archivio negozio o caricato mediante il Pacchetto Loader.

Per informazioni sulla modifica delle risorse del negozio di un archivio negozio esistente, fare riferimento alla guida in linea di WebSphere Commerce. Per

informazioni sulla creazione delle risorse del negozio sotto forma di file XML, fare riferimento alla sezione “Creazione di risorse dati del negozio in un file XML”.

Creazione di risorse dati del negozio in un file XML

Creare le risorse del negozio in formato file XML che possono essere caricati nel database mediante il pacchetto Loader. Se il negozio da creare è multiculturale, è possibile creare file XML diversi per ogni locale supportata dal negozio. Il file specifico delle locale deve contenere tutte le informazioni sulla descrizione, in modo che possa essere facilmente tradotto. Per ulteriori informazioni sul pacchetto Loader, consultare la sezione Parte 7, “Pubblicazione del negozio” a pagina 213.

I negozi di esempio, da cui sono ricavati molti esempi di queste attività, utilizzano un file `store.xml` per tutte le informazioni che non devono essere tradotte e un altro file `store.xml` per ciascuna locale supportata dal negozio, per le informazioni che devono essere tradotte. I file specifici delle locale contengono tutte le informazioni sulla descrizione.

Per creare le risorse del negozio, procedere come segue:

1. Rivedere i file XML utilizzati per creare le risorse del negozio nei negozi di esempio. Tutti i file per i negozi di esempio si trovano nel file dell'archivio negozio corrispondente.

I file dell'archivio negozio si trovano nelle seguenti directory:

-  unità:\WebSphere\CommerceServer\samplestores
-  unità:\Program Files\WebSphere\CommerceServer\samplestores
-  /usr/WebSphere/CommerceServer/samplestores
-  /opt/WebSphere/CommerceServer/samplestores
-  /opt/WebSphere/CommerceServer/samplestores
-  /qibm/proddata/WebCommerce/samplestores

Nota: La guida in linea di WebSphere Commerce contiene le informazioni su ciascuna delle risorse dati contenute nei negozi di esempio.

Ciascun negozio di esempio presenta due file `store.xml`, contenenti le informazioni sul negozio. Per visualizzare i file `store.xml` nell'archivio negozio, decomprimere l'archivio con un programma ZIP. I file `store.xml` risiedono nella directory dei dati. Il file `store.xml` risiede in una sottodirectory specifica per la locale della directory dei dati.

2. Fare riferimento alle informazioni in Appendice B, “Creazione di dati” a pagina 319.
3. Creare un file `store.xml`, copiando uno dei file `store.xml` presenti negli archivi negozio di esempio o creandone uno nuovo. Per ulteriori informazioni, fare riferimento al file DTD corrispondente a `store.xml`. I file DTD si trovano nelle seguenti directory:

-  unità:\WebSphere\CommerceServer\xml\sar
-  unità:\Program Files\WebSphere\CommerceServer\xml\sar
-  /usr/WebSphere/CommerceServer/xml/sar
-  /opt/WebSphere/CommerceServer/xml/sar

-  /opt/WebSphere/CommerceServer/xml/sar
-  /qibm/proddata/WebCommerce/xml/sar

4. Creare un'entità negozio.

- a. Utilizzando il seguente esempio come guida, definire un'entità negozio nel file XML per la tabella STOREENT.

```
<storeent
  storeent_id="@storeent_id_1"
  member_id="&MEMBER_ID"
  type="S"
  identifier="ToolTech"
  setccurr="USD"
/>
```

dove

- storeent_id è una chiave univoca generata.
- member_id è il proprietario dell'entità negozio.
- type è il tipo di entità negozio: G = StoreGroup, S = Store.
- identifier è una stringa che, insieme al proprietario, identifica l'entità negozio in modo univoco.
- setccurr è la valuta predefinita per un'entità negozio, ovvero la valuta che verrà utilizzata da un cliente che non dispone ancora di una valuta preferita. Se è NULL per un negozio, la valuta predefinita viene ottenuta dal relativo gruppo di negozi.

5. Creare un indirizzo del negozio.

- a. Utilizzando il seguente esempio come guida, creare l'indirizzo o gli indirizzi del negozio nel file XML per la tabella STADDRESS. Se il catalogo da creare è multiculturale, è necessario includere queste informazioni in un file XML specifico delle locale.

```
<staddress
  staddress_id="@staddress_id_en_US_1"
  member_id="&MEMBER_ID"
  nickname="storeaddress_English"
  address1="12xx Martindale Avenue"
  address2="Suite 9xx"
  businesstitle="ToolTech"
  city="Toolsville"
  state="Ontario"
  zipcode="Lxx 1xx"
  country="Canada"
  phone1="1-800-555-1234"
  fax1="1-800-555-4321"
  email1="info@tooltech.xxx"
/>
```

dove

- staddress_id è una chiave univoca generata.
- member_id è il proprietario dell'entità negozio.

6. Creare una descrizione per un'entità negozio.

- a. Utilizzando il seguente esempio come guida, creare la descrizione dell'entità negozio nel file XML per la tabella STOREENTDS. Se il catalogo da creare è multiculturale, è necessario includere queste informazioni in un file XML specifico delle locale.

```
<storeentds
  description="Commerce Models Store entity"
  language_id="&en_US"
```

```
displayname="ToolTech"
storeent_id="@storeent_id_1"
staddress_id_cont="@staddress_id_en_US_1"
staddress_id_loc="@staddress_id_en_US_1"
```

dove

- `description` è una descrizione più dettagliata dell'entità negozio, che può essere visualizzata dai clienti.
- `language_id` è la lingua predefinita per le informazioni visualizzate ai clienti che fanno shopping nel negozio.
- `displayname` è una descrizione breve dell'entità negozio, che può essere visualizzata dai clienti.
- `storeent_id` è l'entità negozio.
- `staddress_id_cont` è l'indirizzo di contatto di StoreEntity.
- `staddress_id_loc` è l'ubicazione fisica di StoreEntity.

7. Creare un negozio nel database.

- a. Utilizzando il seguente esempio come guida, definire un negozio nel file XML per la tabella STORE.

```
<store
store_id="@storeent_id_1"
directory="ToolTech"
ffmcenter_id="@ffmcenter_id_1"
language_id="@en_US"
storegrp_id="-1"
allocationgoodfor="43200"
bopmpadfactor="0"
defaultboffset="2592000"
ffmselectionflags="0"
maxboffset="7776000"
rejectedordexpiry="259200"
rtnffmctr_id="@ffmcenter_id_1"
pricerefflags="0"
storetype="B2B"
/>
```

dove

- `store_id` è una chiave univoca generata.
- `directory` è la directory in cui trovare le risorse Web specifiche del negozio. L'ubicazione effettiva di queste risorse nel file system si basa sul valore di questa colonna, oltre ai diversi parametri di configurazione nel file di configurazione di WebSphere Commerce: `StoresDocRoot`, `StoresWebPath` e `StoresPropertiesPath`. Ad esempio, se `StoresDocRoot` è `D:\WebSphere\wcs\stores`, `StoresWebPath` è `web`, `StorePropertiesPath` è `properties` e il valore di questa colonna è `mystore`, i file JSP risiederanno nella directory `D:\WebSphere\wcs\stores\web\mystore` e i file delle proprietà in `D:\WebSphere\wcs\stores\properties\mystore`.
- `ffmcenter_id` è il centro di evasione ordini predefinito per il negozio.
- `language_id` è la lingua predefinita per le informazioni visualizzate ai clienti che fanno shopping nel negozio.
- `storegrp_id` è il gruppo di negozi a cui è associato il negozio. Questo numero viene generato nella tabella STOREGRP.
- `allocationgoodfor` indica che è possibile utilizzare il processo del programma di pianificazione `ReleaseExpiredAllocations` per invertire le assegnazioni dell'inventario ATP alcuni secondi dopo aver effettuato le assegnazioni stesse.

- `bopmpadfactor` indica che se questo negozio calcola gli importi degli ordini (ad esempio, tasse o tariffe di spedizione) in maniera diversa per centri di evasione ordini diversi, l'importo di un ordine precedentemente inoltrato può cambiare quando i centri di evasione ordini vengono infine assegnati agli articoli in sospeso. Questo fattore rappresenta una percentuale di incremento dell'ordine presentato a Payment Manager, se necessario. Ad esempio, specificare 5 per consentire un aumento sino al 5 per cento.
 - `defaultboffset` se non è possibile stabilire un tempo di disponibilità previsto per un ordine in sospeso, in futuro viene impostato su più secondi.
 - `maxboffset` se il tempo della disponibilità stimata per un `OrderItem` messo in backorder in futuro supera questi secondi, esso verrà impostato su tale intervallo di secondi.
 - `rejectedordexpiry` indica il numero di secondi oltre il quale gli ordini con pagamento nello stato respinto sono destinati all'annullamento.
 - `rtnffmctr_id` è il centro di evasione ordini predefinito per la restituzione della merce al negozio.
 - `pricerefflags` contiene gli indicatori che controllano gli accordi commerciali e le offerte ricercati quando i prezzi vengono aggiornati dall'implementazione predefinita del comando di attività `GetContractUnitPrices`:
 - 1 = `usePreviousOnly` - Utilizza quelli menzionati da `OrderItems`. Se non è più possibile utilizzarli, non vengono ricercati.
 - 2 = `usePreviousOrSearchAgain` - Simile a `usePreviousOnly`, ma al contrario, quando non è più possibile utilizzarli, ricerca quelli salvati nelle tabelle `ORDIOFFER` e `ORDITRD`
 - 4 = `alwaysSearchAgain` - Cerca sempre quelli salvati nelle tabelle `ORDIOFFER` e `ORDITRD`.
 - `storetype` indica uno dei seguenti tipi di negozio, per l'utilizzo da parte di un'interfaccia utente che fornisce le funzioni appropriate a seconda dello `StoreType`: B2B = Business-to-Business. B2C = Business-to-Consumer.
8. Definire una lingua supportata per il negozio.

- a. Utilizzando il seguente esempio come guida, definire una lingua supportata per il negozio nel file XML aggiungendo informazioni alla tabella `STORELANG`. Se il negozio supporta più lingue, inserire queste informazioni in un file specifico delle locale (uno per ciascuna lingua supportata dal negozio).

```
<storelang
  language_id="&en_US"
  storeent_id="@storeent_id_1"
/>
```

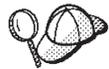
dove

- `language_id` è la lingua supportata dall'entità negozio.
 - `storeent_id` è l'entità negozio.
- b. Utilizzando il seguente esempio come guida, aggiungere informazioni sulla lingua alla tabella `STORELANGDS`. Se il negozio supporta più lingue, inserire queste informazioni in un file specifico delle locale (uno per ciascuna lingua supportata dal negozio).

```
<storlangds
  description="United States"
  language_id="&en_US"
  storeent_id="@storeent_id_1"
  language_id_desc="& en_US"
/>
```

dove

- `description` è una breve descrizione della lingua, che può essere visualizzata dai clienti in un elenco di selezione.
- `language_id` è la lingua della descrizione.
- `storeent_id` è l'entità negozio che supporta la lingua.
- `language_id_desc` è la lingua da descrivere.



Per ulteriori informazioni sull'uso di @ e &, fare riferimento all'Appendice B, "Creazione di dati" a pagina 319.

Capitolo 7. Dati di registro comandi, viste e URL

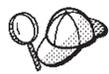
Quando si crea un'istanza di WebSphere Commerce Server, i comandi, le viste e gli URL predefiniti forniti con WebSphere Commerce vengono registrati nel database di WebSphere Commerce Server nelle tabelle corrispondenti: CMDREG, VIEWREG e URLREG. Questi comandi, viste e URL sono disponibili in tutti i negozi che risiedono nell'istanza.

WebSphere Commerce fornisce inoltre file JSP predefiniti per aprire le viste predefinite. I file JSP sono associati alle viste nella tabella VIEWREG.

Se si creano nuovi comandi, viste o URL o si personalizzano quelli esistenti, è necessario registrarli nelle tabelle di database corrispondenti (CMDREG, VIEWREG e URLREG) perché siano disponibili nel negozio. Se si creano nuovi file JSP per il negozio, è necessario associare tali file alla visualizzazione corrispondente nella tabella VIEWREG.

Nota: Se si crea un nuovo file JSP, ma si attribuisce a questo file lo stesso nome del file JSP predefinito associato alla visualizzazione, non è necessario registrare il nuovo file nella tabella VIEWREG.

Per ulteriori informazioni sui comandi di creazione o personalizzazione, le viste o gli URL, consultare il manuale *IBM WebSphere Commerce - Guida per il programmatore*. La *Guida per il programmatore* contiene anche informazioni su come e quando registrare comandi, viste, URL e file JSP.



Per informazioni più dettagliate sulla struttura delle risorse comandi e viste di WebSphere Commerce Server, fare riferimento ai modelli di dati di comandi e viste della guida in linea di WebSphere Commerce.

Registrazione di comandi, viste e URL in WebSphere Commerce

Se si creano o si personalizzano nuovi comandi, viste, URL o file JSP per il negozio, è possibile registrare tali elementi mediante un file XML, da caricare nel database con il Pacchetto Loader o come parte di un archivio negozio da pubblicare mediante Servizi del negozio. Per ulteriori informazioni sul pacchetto Loader, consultare la Parte 7, "Pubblicazione del negozio" a pagina 213.

Nota: Prima di creare un file XML per caricare comandi nuovi o personalizzati, fare riferimento alla *Guida per il programmatore* per altri dettagli sul funzionamento dei comandi.

Creazione di un file XML per registrare comandi, viste e URL

Per creare un file XML per registrare nuovi comandi, viste e file JSP per il negozio, procedere come segue:

1. Rivedere i file XML utilizzati per registrare i comandi, le viste, i file JSP per i negozi di esempio. Ogni negozio di esempio presenta un file `command.xml`, che comprende informazioni sulla registrazione. I file dell'archivio negozio si trovano nelle seguenti directory:

-  unità: \WebSphere\CommerceServer\samplstores

- ▶ 2000 unità:\Program Files\WebSphere\CommerceServer\samplstores
- ▶ AIX /usr/WebSphere/CommerceServer/samplstores
- ▶ Solaris /opt/WebSphere/CommerceServer/samplstores
- ▶ Linux /opt/WebSphere/CommerceServer/samplstores
- ▶ 400 /qibm/proddata/WebCommerce/samplstores

Nota: La guida in linea di WebSphere Commerce contiene le informazioni su ciascuna delle risorse dati contenute nei negozi di esempio.

Per visualizzare il contenuto dell'archivio negozio, utilizzare un programma di decompressione. Il file `command.xml` risiede nella directory `data`.

- Fare riferimento alle informazioni nell'Appendice B, "Creazione di dati" a pagina 319.
- Creare un file `command.xml` copiando uno dei file `command.xml` presenti negli archivi negozio di esempio o creandone uno nuovo. Per ulteriori informazioni, fare riferimento al file DTD corrispondente a `command.xml`. I file DTD si trovano nelle seguenti directory:

- ▶ NT unità:\WebSphere\CommerceServer\xml\sar
- ▶ 2000 unità:\Program Files\WebSphere\CommerceServer\xml\sar
- ▶ AIX /usr/WebSphere/CommerceServer/xml/sar
- ▶ Solaris /opt/WebSphere/CommerceServer/xml/sar
- ▶ Linux /opt/WebSphere/CommerceServer/xml/sar
- ▶ 400 /qibm/proddata/WebCommerce/xml/sar

- I comandi del controller devono essere registrati nella tabella URLREG e nella tabella CMDREG. Per registrare un comando del controller nuovo o personalizzato nella tabella URLREG, creare un'entrata nel file XML per ciascun nuovo comando personalizzato del controller, utilizzando il seguente esempio come guida:

```
<urlreg
url="MyProductDisplay"
storeent_id="@storeent_id_1"
interfacename="com.mystore.commerce.catalog.commands.ProductDisplayCmd"
https="0"
description="Product display command for my store"
authenticated="0"
internal="0" />
dove
```

- `urlreg` è il nome della tabella di database contenente queste informazioni.
- `url` è il nome dell'URL
- `storeent_id` è l'ID dell'entità negozio e l'utilizzo del simbolo `@` è riconosciuto come risoluzione dell'alias interno. Quando si utilizzano le risoluzioni dell'alias interno, un alias sostituisce la chiave principale (identificativo) nel documento XML. L'alias viene quindi utilizzato altrove nel file XML per indicare tale elemento. Questo elimina la necessità di conoscere gli indici univoci necessari per creare il file XML. Durante la

pubblicazione, ID Resolver sostituisce il simbolo @ con un valore univoco. Per ulteriori informazioni, consultare l'Appendice B, "Creazione di dati" a pagina 319.

- `interfacename` è il nome dell'interfaccia del comando del controller
 - `https` è l'HTTP sicuro obbligatorio per questa richiesta di URL. Utilizzare 1 quando l'HTTP sicuro è obbligatorio e 0 quando non lo è.
 - `authenticated` indica se il collegamento utente è obbligatorio o meno per questa richiesta di URL. Utilizzare 1 quando l'autenticazione è obbligatoria e 0 quando non lo è.
 - `internal` indica se il comando è interno a WebSphere Commerce. Gli URL interni sono utilizzati dagli strumenti di WebSphere Commerce. Utilizzare 1 se la visualizzazione è interna e 0 se è esterna. Gli URL creati devono essere esterni.
5. Per registrare un nuovo comando del controller o di attività nella tabella `CMDREG`, creare un'entrata nel file XML per ciascun comando del controller o di attività nuovo o personalizzato, utilizzando il seguente esempio di comando di attività (dal file `command.xml` del negozio di esempio ToolTech) come guida:

```
< cmdreg
storeent_id="@storeent_id_1"
interfacename="com.ibm.commerce.payment.commands.DoPaymentCmd"
classname="com.ibm.commerce.payment.commands.DoPaymentMPFCmdImpl"/>
dove
```

- `cmdreg` è il nome della tabella di database (`CMDREG`) contenente queste informazioni.
 - `storeent_id` è l'ID dell'entità negozio e l'utilizzo del simbolo @ è riconosciuto come risoluzione dell'alias interno. Quando si utilizzano le risoluzioni dell'alias interno, un alias sostituisce la chiave principale (identificativo) nel documento XML. L'alias viene quindi utilizzato altrove nel file XML per indicare tale elemento. Questo elimina la necessità di conoscere gli indici univoci necessari per creare il file XML. Durante la pubblicazione, ID Resolver sostituisce il simbolo @ con un valore univoco. Per ulteriori informazioni, consultare l'Appendice B, "Creazione di dati" a pagina 319.
 - `interfacename` è il nome dell'interfaccia del comando.
 - `classname` è il nome della classe di implementazione del comando. Generalmente, si tratta del nome dell'interfaccia con `Impl` alla fine.
6. Per registrare nuove visualizzazioni o per associare nuovi file JSP ad una visualizzazione, creare un'entrata nella tabella `VIEWREG`, utilizzando il seguente esempio (dal file `command.xml` del negozio di esempio ToolTech) come guida:

```
<viewreg
viewname="OrderOptionsView"
devicefmt_id="-1"
storeent_id="@storeent_id_1"
interfacename="com.ibm.commerce.command.ForwardViewCommand"
classname="com.ibm.commerce.command.HttpForwardViewCommandImpl"
properties="docname=Shipping.jsp"
internal="0"
https="0"/>
dove
```

- viewreg è il nome della tabella di database (VIEWREG) contenente queste informazioni.
- viewname è il nome della vista.
- devicefmt_id è il tipo di unità su cui verrà utilizzata questa visualizzazione, ad esempio un browser.
- storeent_id è l'ID dell'entità negozio e l'utilizzo del simbolo @ è riconosciuto come risoluzione dell'alias interno. Quando si utilizzano le risoluzioni dell'alias interno, un alias sostituisce la chiave principale (identificativo) nel documento XML. L'alias viene quindi utilizzato altrove nel file XML per indicare tale elemento. Questo elimina la necessità di conoscere gli indici univoci necessari per creare il file XML. Durante la pubblicazione, ID Resolver sostituisce il simbolo @ con un valore univoco. Per ulteriori informazioni, consultare l'Appendice B, "Creazione di dati" a pagina 319.
- interfacename è il nome dell'interfaccia del comando di visualizzazione. Le opzioni predefinite sono ForwardView, DirectView e RedirectView.
- classname è il nome della classe di implementazione della visualizzazione. Generalmente, si tratta del nome dell'interfaccia con Impl alla fine.
- properties è la coppia predefinita nome valore impostata come proprietà di immissione per il comando. Se viene visualizzata sempre la stessa pagina, impostare il nome file JSP in questa proprietà, ad esempio, docname=Shipping.jsp.
- internal indica se la visualizzazione è interna a WebSphere Commerce. Le visualizzazioni esterne vengono utilizzate dagli strumenti WebSphere Commerce. Utilizzare 1 se la visualizzazione è interna e 0 se è esterna. Le visualizzazioni create devono essere esterne.
- https è l'HTTP sicuro obbligatorio per questa richiesta di URL. Utilizzare 1 quando l'HTTP sicuro è obbligatorio e 0 quando non lo è.



Per ulteriori informazioni sull'uso di @ e &, fare riferimento all'Appendice B, "Creazione di dati" a pagina 319.

Capitolo 8. Risorse catalogo

Come un catalogo tradizionale, il catalogo in linea è costituito da merci e servizi offerti per la vendita. Nonostante le dimensioni e la struttura dei cataloghi in linea possano differire enormemente da negozio a negozio, a seconda del tipo e della quantità di merci disponibili per l'acquisto, per i cataloghi sono necessari i seguenti elementi:

- Merci in vendita, inclusi
 - Prezzi, quasi sempre compresi nel catalogo in linea.
 - Dati dei prodotti, quali descrizioni e immagini delle merci.
 - Categorie, per semplificare ai clienti l'esplorazione, dal momento che molti cataloghi, ma non tutti, dividono le merci in categorie.
- Metodo di visualizzazione per il prodotto in vendita. Le pagine di visualizzazione del catalogo indicano il modo in cui una pagina verrà visualizzata ai clienti e forniscono coerenza alle varie pagine del catalogo. Il modo in cui il catalogo viene strutturato dipende dalla merce del negozio.

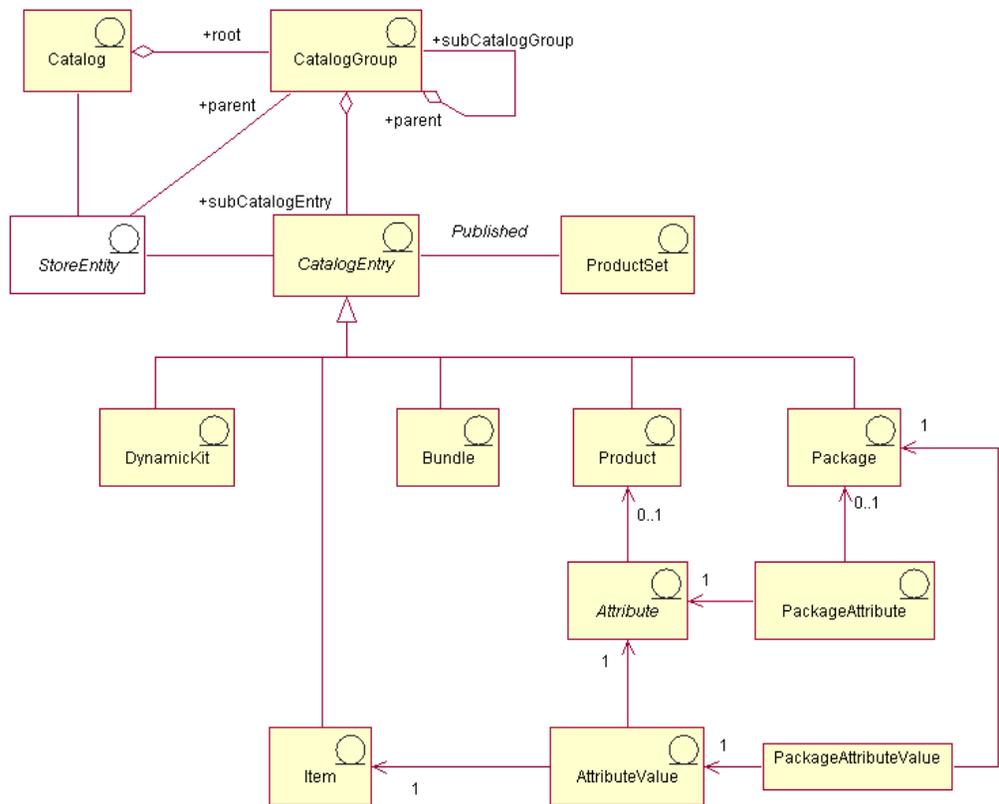
Cataloghi in WebSphere Commerce

WebSphere Commerce pone vari requisiti per i cataloghi in linea del negozio. Ciascun negozio nel sistema WebSphere Commerce deve disporre di un catalogo principale. Il catalogo principale è la posizione centrale da cui gestire la merce del negozio. E' il singolo catalogo che contiene tutti i prodotti, articoli, relazioni e prezzi standard per tutto ciò che è in vendita nel negozio.

È possibile condividere il catalogo principale tra i negozi e definire tutti i negozi necessari. Oltre alla creazione di un catalogo principale per la gestione del catalogo, è possibile anche scegliere di creare uno o più cataloghi di esplorazione a scopi di visualizzazione. Un catalogo di esplorazione può contenere le stesse voci del catalogo principale, ma con una struttura più flessibile per la visualizzazione ai clienti. E' possibile avere tutti i cataloghi di esplorazione desiderati. Tuttavia, poiché per gestire la merce in linea viene utilizzato il catalogo principale, si consiglia di utilizzare tale catalogo anche come catalogo di esplorazione, per minimizzare il costo gestionale.

Se si crea un nuovo catalogo principale da utilizzare con un negozio di WebSphere Commerce o si utilizza un catalogo principale esistente disponibile in un negozio di esempio di WebSphere Commerce, ad esempio ToolTech, sarà necessario modificare il catalogo in base a tali requisiti. Il grafico di seguito riportato illustra

la struttura di base di un catalogo in WebSphere Commerce.



Questo grafico, insieme a tutti gli altri grafici della sezione relativa ai dati del negozio fa parte del modello di informazioni di WebSphere Commerce. Per ulteriori informazioni sul modello di informazioni, fare riferimento alla sezione “Modello di informazioni dati del negozio” a pagina 27. Per ulteriori informazioni sulle convenzioni utilizzate in questo grafico, fare riferimento all’Appendice A, “Legenda UML” a pagina 317.

Cataloghi

Il *catalogo* è il punto di partenza per il modello di informazioni. Il catalogo contiene tutte le informazioni gerarchiche e per l’esplorazione relative al catalogo in linea e rappresenta un insieme di gruppi di catalogo o articoli di catalogo disponibile per la visualizzazione e l’acquisto presso un negozio in linea.

In WebSphere Commerce, un catalogo è rappresentato nel database da un’*entità catalogo*. Un’entità catalogo è costituita da un ID catalogo e da una descrizione del catalogo, ad esempio, il nome del catalogo. Dal momento che ciascun catalogo rappresenta un’entità univoca e diversa, può essere associato a uno o più negozi. Ciascun negozio nel sistema WebSphere Commerce deve essere collegato ad almeno un’entità catalogo. Il catalogo principale è un catalogo speciale che contiene tutti gli articoli in vendita nel negozio in linea.

Gruppi di catalogo

I *gruppi di catalogo* sono raggruppamenti generici di voci di catalogo, creati per scopi esplorativi e di divisione dei cataloghi. Un gruppo di catalogo appartiene ad

un catalogo e può contenere più di un gruppo di catalogo o di voci di catalogo. E' possibile associare gruppi di catalogo a più di un catalogo. Un gruppo di catalogo è conosciuto anche come *categoria*.

Un catalogo piatto non raggruppa i prodotti in categorie, ma visualizza un elenco di prodotti. Nonostante sia possibile creare in WebSphere Commerce un catalogo piatto, si consiglia di creare nel catalogo dei gruppi per scopi strutturali e esplorativi.

Quando si creano gruppi di catalogo, è necessario disporre il catalogo in una gerarchia o albero invertito. L'albero inizia con un gruppo di catalogo generico (denominato categoria radice) e si allarga in sottocategorie sempre più specifiche fino a quando non può essere diviso ulteriormente. Ogni gruppo di catalogo di livello inferiore, contenente solo prodotti, rappresenta una foglia. Un gruppo di catalogo è principale rispetto alle categorie inferiori e secondario rispetto a quelle superiori. Ad esempio, Moda uomo è un raggruppamento di categorie di accessori maschili, mentre i gruppi di catalogo Pantaloni e Camicie sono raggruppamenti di prodotti.

Voci di catalogo

Ciascun gruppo di catalogo contiene *voci di catalogo*. Le voci di catalogo sono gli articoli contenuti in un catalogo in linea. Normalmente dispongono di un nome o di un numero di parte, di una descrizione, di uno o più prezzi, di immagini e di altri dettagli. Una voce di catalogo può essere un prodotto, un articolo, un pacco, un bundle o un kit dinamico. Se necessario, è possibile creare nuovi tipi di voci di catalogo che non rientrano in uno dei cinque modelli esistenti. Ulteriori informazioni su ciascun tipo di voce di catalogo sono di seguito riportate.

Prodotti

Un *prodotto* è un tipo di voce di catalogo. Un prodotto agisce come modello per un gruppo di articoli o SKU che presentano gli stessi attributi. Ad esempio, una camicia rappresenta un prodotto del catalogo. Una volta aggiunti attributi e valori di attributi alla camicia, ogni variazione diventa un articolo, ad esempio una camicia piccola nera.

Articoli

Un *articolo* è un'unità concreta di merce che ha un nome specifico, un numero parte e un prezzo. Ad esempio, una camicia piccola nera è un articolo mentre una camicia è un prodotto. Tutte le SKU relative ad un determinato prodotto dispongono dello stesso insieme di attributi e si distinguono mediante i valori degli attributi.

Nota: Per gli utenti di WebSphere Commerce Accelerator, articoli e *SKU* sono considerati sinonimi. Quando si utilizzano gli strumenti di gestione dei prodotti in WebSphere Commerce Accelerator, gli articoli che è possibile ordinare vengono denominati SKU. Nello schema di database di WebSphere Commerce, questo particolare tipo di voce di catalogo è definito articolo.

Pacchetti

Un *pacchetto* è un insieme indivisibile di voci di catalogo. Ad esempio, un pacchetto computer può contenere un'unità di elaborazione, un video e un'unità disco fisso che non possono essere venduti separatamente. Simile a un prodotto, un pacchetto presenta attributi di definizione ed è un contenitore per i pacchetti scomposti. Un pacchetto scomposto è paragonabile ad una SKU. Il pacchetto, a cui

viene assegnato un prezzo, rappresenta una SKU ordinabile e può essere aggiunto a un carrello degli acquisti. Durante l'esplorazione o dopo averlo aggiunto al carrello, non è possibile scomporre o modificare il pacchetto.

Bundle

Un *bundle* è un insieme di voci di catalogo che consente ai clienti di comprare più articoli contemporaneamente. Ad esempio, un bundle per un computer potrebbe essere composta da un'unità centrale di elaborazione, un video, un disco fisso e un'unità CD-ROM. Un bundle è un raggruppamento di articoli o una combinazione di prodotti, articoli e pacchetti scomposti. Se viene selezionato un bundle che contiene solo articoli, viene scomposto in SKU ordinabili separatamente che vengono aggiunti singolarmente al carrello degli acquisti. Tuttavia, se viene selezionato un bundle che contiene prodotti, questi prodotti devono essere trasformati in articoli tramite la risoluzione SKU prima di poterli aggiungere al carrello degli acquisti. In entrambi i casi, una volta scomposto il bundle e aggiunti gli articoli che lo compongono ad un carrello degli acquisti, è possibile modificare o rimuovere ciascun articolo.

Kit dinamici

Un *kit dinamico* è un tipo di voce di catalogo che può essere configurato dinamicamente dal cliente. Questa configurazione (o raggruppamento) di prodotti è basata sulle esigenze del cliente e viene venduta come unità singola. I componenti di un kit dinamico vengono composti da un programma di configurazione di prodotti esterno, attraverso un insieme di regole predefinite e l'interazione dell'utente. L'aggiunta di un kit dinamico ad un ordine è analoga a quella di un pacchetto. Come accade per un pacchetto, non è possibile modificare i singoli componenti di un kit dinamico ed è necessario completare l'intera configurazione come insieme. Tuttavia, è possibile modificare i componenti del kit dinamico riconfigurandolo mediante un programma di configurazione del prodotto esterno.

Gruppi di prodotti

I *gruppi di prodotti* sono associati agli oggetti CatalogEntry pubblicati. I gruppi di prodotti forniscono un meccanismo con cui è possibile dividere il catalogo in sottoinsiemi logici. Questa suddivisione consente di mostrare parti diverse del catalogo a utenti diversi. È possibile creare un contratto e specificare che ai partecipanti di tale contratto è solo concesso di acquistare prodotti che rientrano in un insieme di prodotti predefinito. WebSphere Commerce fornisce strumenti per la creazione di insiemi di prodotti dal catalogo principale, che è possibile utilizzare nei contratti per il filtraggio delle concessioni.

Attributi

Gli *attributi* sono proprietà di prodotti in un negozio in linea, come colore o dimensione. Un attributo fa parte di un prodotto. Ciascuna combinazione possibile di attributi e valori di attributo definisce un articolo.

Valori di attributo

I *valori di attributo* sono le proprietà di un attributo, quali un determinato colore (blu o giallo) o la taglia (piccola, media o grande). È necessario definire i valori di attributo prima di assegnare tali valori agli articoli. Ciascuna combinazione possibile di attributi e valori di attributo definisce un articolo.

Attributi di pacchetto

Gli *attributi di pacchetto* sono ereditati dagli attributi dei prodotti contenuti nei pacchetti. Un pacchetto può non avere attributi di pacchetto o avere più attributi di pacchetto. Un attributo di pacchetto fa riferimento ad un solo attributo.

Valori degli attributi di pacchetto

I *valori degli attributi di pacchetto* sono i valori assegnati agli attributi di pacchetto. I valori degli attributi di pacchetto sono ereditati dai valori degli attributi dei prodotti contenuti nei pacchetti. Il valore di un attributo di pacchetto fa riferimento ad un solo valore di attributo.



Per informazioni più dettagliate sulla struttura delle risorse catalogo in WebSphere Commerce, fare riferimento ai modelli di dati di catalogo nella guida in linea di WebSphere Commerce.

Creazione di risorse catalogo in WebSphere Commerce

Per creare le risorse catalogo per il negozio, è necessario creare un catalogo principale aggiungendo informazioni a varie tabelle di database di WebSphere Commerce. E' possibile creare il catalogo mediante file XML caricati nel database utilizzando il pacchetto Loader. Se il catalogo da creare è multiculturale, sono necessari file XML separati per ogni locale supportata dal negozio. Ciascun file XML specifico della locale aggiunge informazioni traducibili, quali descrizioni per il catalogo, i gruppi di catalogo e le voci di catalogo.

Di seguito è riportata una panoramica del processo di creazione del catalogo:

1. In WebSphere Commerce, un catalogo viene creato mediante file XML. La creazione di un catalogo inizia con un'entità catalogo, l'equivalente del database del catalogo in carta.
2. Creare la funzione di navigazione e la struttura del catalogo aggiungendo i gruppi di catalogo in modo da determinare le categorie e la disposizione della merce.
3. Creare informazioni di inventario come base per le voci del catalogo.
4. Aggiungere la merce come voci di catalogo rappresentanti prodotti, SKU, pacchi, raccolte di prodotti e kit dinamici.
5. Gli attributi e i relativi valori vengono aggiunti ai prodotti del catalogo per distinguere le diverse SKU.
6. È possibile creare pacchi e raccolte di prodotti per raggruppare alcune voci di catalogo per scopi promozionali.
7. Le relazioni tra i gruppi di catalogo e le voci di catalogo vengono create successivamente. Tali relazioni determinano quali voci appartengono a un gruppo di catalogo.
8. È possibile commercializzare le associazioni per le voci di catalogo come strategie di raccomandazione di prodotti.
9. Associare il catalogo, i gruppi di catalogo e le voci di catalogo al negozio WebSphere Commerce.
10. Nelle operazioni finali, è necessario creare:
 - a. tasse per la merce,
 - b. metodi di spedizione,

- c. un centro di evasione ordini che funga da magazzino di inventario e un centro di spedizione e ricezione ordini (è possibile definire più di un centro di evasioni ordini per un negozio),
- d. prezzi per la merce.

Creazione di un catalogo principale

Per creare un catalogo principale contenente più livelli di categorie, procedere come segue:

Parte 1: Preparazione della creazione del catalogo

1. Rivedere le informazioni sul catalogo e i modelli di oggetti e dati corrispondenti in WebSphere Commerce. Le informazioni sul catalogo rappresentano un componente di WebSphere Commerce Server che consente l'esplorazione in linea, la suddivisione in categorie e la suddivisione in associazioni del catalogo in linea per la merce ordinabile.
2. Rivedere le informazioni sul pacchetto WebSphere Commerce Loader. Il pacchetto Loader è composto principalmente da programmi di utilità per la preparazione e il caricamento dei dati in un database di WebSphere Commerce. E' possibile utilizzare il pacchetto Loader per caricare una grossa quantità di dati e per aggiornare i dati nel database. Per ulteriori informazioni sul pacchetto Loader, consultare la Parte 7, "Pubblicazione del negozio" a pagina 213.
3. Fare riferimento alle informazioni nell'Appendice B, "Creazione di dati" a pagina 319.
4. Creare un'organizzazione mediante la console di gestione che agisca come proprietario del catalogo. Per ulteriori informazioni, fare riferimento alla sezione "Creazione di un'organizzazione" nella guida in linea di WebSphere Commerce.
5. Creare un nuovo file XML per il catalogo principale utilizzando le voci XML esistenti ed i file `catalog.xml` del negozio di esempio ToolTech come guida. Se il catalogo da creare è multiculturale, creare un file `catalog.xml` diverso per ogni locale supportata dal negozio. Il file specifico delle locale deve contenere tutte le informazioni sulla descrizione, in modo che possa essere facilmente tradotto. In quest'esempio, un file `catalog.xml` verrà utilizzato per tutte le informazioni che non devono essere tradotte e un altro file `catalog.xml` per ciascuna locale supportata dal negozio, per le informazioni che devono essere tradotte. In alternativa, se si preferisce, è possibile utilizzare il file XML esistente del negozio di esempio ToolTech e modificare le informazioni in base alle necessità. I file `catalog.xml` del negozio di esempio ToolTech risiedono nel file di archivio negozio. Per visualizzare i file `catalog.xml`, decomprimere l'archivio negozio mediante un programma ZIP. I file `catalog.xml` si trovano nelle seguenti directory di dati:

-  `unità:\WebSphere\CommerceServer\samplstores`
-  `unità:\Program Files\WebSphere\CommerceServer\samplstores`
-  `/usr/WebSphere/CommerceServer/samplstores`
-   `/opt/WebSphere/CommerceServer/samplstores`
-  `/qibm/proddata/WebCommerce/samplstores`

Nota: La guida in linea di WebSphere Commerce contiene le informazioni su ciascuna delle risorse dati contenute nei negozi di esempio.

Il file catalog.dtd si trova nella seguente directory:

-  *unità*: \WebSphere\CommerceServer\xml\sar
-  *unità*: \Program Files\WebSphere\CommerceServer\xml\sar
-  /usr/WebSphere/CommerceServer/xml/sar
-   /opt/WebSphere/CommerceServer/xml/sar
-  /qibm/proddata/WebCommerce/xml/sar

Parte 2: Creazione di un'entità catalogo

1. Utilizzando come guida l'esempio di seguito riportato del negozio ToolTech, creare un'entità catalogo aggiungendo informazioni alle tabelle CATALOG e CATALOGDSC. Un'entità del catalogo rappresenta un catalogo nel database.

```
<catalog
catalog_id="@catalog_id_1"
member_id("&MEMBER_ID;"
identifier="ToolTech"
description="ToolTech Catalog"
tpclevel="0"
/>
```

dove

- catalog_id è il numero di riferimento interno.
 - member_id è il numero di riferimento interno che identifica il proprietario del catalogo.
 - identifier è un nome esterno per il catalogo.
 - description è una descrizione del catalogo.
2. Utilizzando come guida l'esempio di seguito riportato del negozio ToolTech, aggiungere la descrizione del catalogo nel file XML specifico delle locale a scopi di conversione:

```
<catalogdsc
catalog_id="@catalog_id_1"
language_id("&en_US;"
name="Store master catalog"
/>
```

dove

- catalog_id è il numero di riferimento interno che collega informazioni specifiche della lingua ad un catalogo.
- language_id è l'identificativo della lingua.
- name è il nome del catalogo dipendente dalla lingua.

Parte 3: Creazione di gruppi di catalogo

1. Utilizzando come guida l'esempio di seguito riportato del negozio ToolTech, creare gruppi di catalogo aggiungendo informazioni alle tabelle CATGROUP e CATGRPDESC. I gruppi di catalogo, conosciuti anche come categorie, sono raggruppamenti di altri gruppi di catalogo o prodotti. Completare questa attività per ciascun gruppo di catalogo presente nel catalogo:

```
<catgroup
catgroup_id="@catgroup_id_1"
member_id("&MEMBER_ID;"
identifier="Woodworking"
markfordelete="0"
/>
```

dove

- `catgroup_id` è il numero di riferimento interno del gruppo di catalogo
- `member_id` è il numero di riferimento interno che identifica il proprietario del catalogo.
- `identifer` è un nome esterno per il catalogo.
- `markfordelete` indica se il gruppo di catalogo è stato contrassegnato per la cancellazione:
 - 0 = No.
 - 1 = Sì.

2. Utilizzando come guida l'esempio di seguito riportato del negozio ToolTech, aggiungere la descrizione del gruppo di catalogo nel file XML specifico delle locale a scopi di conversione. Completare questa attività per ciascun gruppo di catalogo presente nel catalogo:

```
<catgrpdesc
language_id="&en_US;"
catgroup_id="@catgroup_id_1"
name="Woodworking"
shortdescription="Woodworking"
longdescription="Woodworking"
published="1"
/>
```

dove

- `language_id` è l'identificativo della lingua.
- `catgroup_id` è il numero di riferimento interno del gruppo di catalogo.
- `name` è il nome che dipende dalla lingua del catalogo.
- `shortdescription` è una descrizione breve del gruppo di catalogo.
- `longdescription` è una descrizione dettagliata del gruppo di catalogo.
- `published` indica se questo gruppo di catalogo deve essere visualizzato per la lingua indicata da `language_id`:
 - 0 = No.
 - 1 = Sì.

Nota: Ogni volta che si crea un gruppo di catalogo con la relativa descrizione, la sequenza di `catgroup_id` varia per rappresentare un nuovo gruppo di catalogo. Ad esempio, `catgroup_id="@catgroup_id_2"`, `catgroup_id="@catgroup_id_3"` e `catgroup_id="@catgroup_id_4"` e così via.

3. Una volta creati i gruppi di catalogo, assegnare un gruppo di catalogo di livello superiore al catalogo, aggiungendo informazioni alla tabella CATTGRP. Questo gruppo di catalogo è principale rispetto ai gruppi di catalogo immediatamente inferiori. Completare questa attività per ciascun gruppo di catalogo di livello superiore presente nel catalogo. Utilizzare come guida l'esempio di seguito riportato del negozio ToolTech:

```
<cattgrp
catalog_id="@catalog_id_1"
catgroup_id="@catgroup_id_1"
/>
```

dove

- `catalog_id` è il numero di riferimento del catalogo.
- `catgroup_id` è il numero di riferimento del gruppo di catalogo.

Nota: Ogni volta che si assegnano gruppi di catalogo di livello superiore al catalogo, la sequenza di `catgroup_id` varia per rappresentare una nuova associazione di gruppi di catalogo. Ad esempio, `catgroup_id="@catgroup_id_2"`, `catgroup_id="@catgroup_id_3"` e `catgroup_id="@catgroup_id_4"` e così via.

- Una volta che la struttura principale e secondaria è stata determinata per i gruppi di catalogo, creare relazioni tra i gruppi di catalogo aggiungendo informazioni alla tabella `CATGRPREL`. Completare questa attività per ciascuna struttura di gruppi di catalogo principali e secondari presenti nel catalogo. Utilizzare come guida l'esempio di seguito riportato del negozio ToolTech:

```
<catgrprel  
catgroup_id_parent="@catgroup_id_1"  
catgroup_id_child="@catgroup_id_11"  
catalog_id="@catalog_id_1"  
sequence="0"  
>
```

dove

- `catgroup_id_parent` è il gruppo di catalogo di origine di questa relazione.
- `catgroup_id_child` è il gruppo di catalogo di destinazione di questa relazione.
- `catalog_id` è il numero di riferimento del catalogo.
- `sequence` è il numero che determina l'ordine di visualizzazione del contenuto del gruppo di catalogo.

Nota: Con ciascuna relazione tra i gruppi di catalogo, le sequenze di `catgroup_id_child` e `sequence` variano per rappresentare una nuova relazione. Ad esempio, le relazioni successive verranno visualizzate come `catgroup_id_child="@catgroup_id_12"` e `sequence="1"` e `catgroup_id_child="@catgroup_id_13"` e `sequence="2"` e così via. Se non si utilizza una struttura di navigazione nel catalogo, è possibile rimuovere la relazione `CATGRPREL`.

Parte 4: Creazione delle informazioni di inventario

- Utilizzando come guida l'esempio di seguito riportato del negozio ToolTech, creare le informazioni sull'inventario aggiungendo informazioni alle tabelle `BASEITEM`, `BASEITEMDSC`, `ITEMSPC`, `ITEMVERSN`, `VERSIONSPC`, `DISTARRANG` e `STOREITEM`. Iniziare creando gli articoli di base aggiungendo informazioni alla tabella `BASEITEM`. Gli articoli di base rappresentano un gruppo generale di prodotti con un nome e una descrizione comune. Completare questa attività per ciascun gruppo di articoli di inventario presente nel catalogo:

```
<baseitem  
baseitem_id="@baseitem_id_102"  
member_id="@MEMBER_ID;"  
markfordelete="0"  
partnumber="toolttech_sku_102"  
itemtype_id="ITEM"  
quantitymeasure="C62"  
quantitymultiple="1.0"  
>
```

dove

- `baseitem_id` è la chiave univoca generata.
- `member_id` è il proprietario dell'articolo di base.

- `markfordelete` indica se l'articolo di base è stato contrassegnato per la cancellazione:
 - 0 = No.
 - 1 = Sì.
- `partnumber` identifica in modo univoco l'articolo di base per il proprietario.
- `itemtype_id` è il tipo di articolo di base:
 - ITEM = articoli, pacchetti o bundle
 - DNKT = kit dinamici.
- `quantitymeasure` è l'unità di misura per multipli di quantità.
- `quantitymultiple` è l'importo dell'articolo di base calcolato in unità integrali. Insieme a `quantitymeasure`, indica quanto rappresenta ciascuna unità integrale.

Nota: E' necessario creare un articolo di base per ciascun prodotto che si desidera creare nel catalogo. Ogni volta che si crea un articolo di base, i numeri `baseitem_id` e `partnumber` variano per creare un nuovo articolo di base. Ad esempio, un articolo di base conterrà le voci `baseitem_id="@baseitem_id_147"` e `partnumber="tooltech_sku_147"`, mentre un altro articolo di base conterrà le voci `baseitem_id="@baseitem_id_192"` e `partnumber="tooltech_sku_192"` e così via.

2. Utilizzando come guida l'esempio di seguito riportato del negozio ToolTech, aggiungere informazioni sugli articoli specificati al database. Un articolo specificato è un articolo con un valore per ogni relativo attributo e rappresenta un articolo, un pacchetto, un bundle o un kit dinamico del catalogo. Completare questa attività per ciascun articolo specificato presente nel catalogo:

```
<itemspc
itemspc_id="@itemspc_id_106"
baseitem_id="@baseitem_id_102"
markfordelete="0"
partnumber="T0000106"
member_id="MEMBER_ID;"
discontinued="N"
/>
```

dove

- `itemspc_id` è la chiave univoca generata.
- `baseitem_id` è l'articolo di base del prodotto.
- `markfordelete` indica se l'articolo specificato è stato contrassegnato per la cancellazione:
 - 0 = No.
 - 1 = Sì.
- `partnumber` identifica in modo univoco l'articolo specificato per il proprietario.
- `member_id` è il proprietario dell'articolo specificato.
- `discontinued` indica se l'articolo specificato è fuori produzione:
 - Y=E' fuori produzione e può essere ordinato se vi è merce sufficiente ma non può essere messo in sospenso.
 - N=E' ancora in produzione e può essere messo in sospenso se non è disponibile in magazzino.

Nota: E' necessario creare un articolo specificato per ciascun articolo che si desidera creare nel catalogo. Ogni volta che si definisce un articolo specificato, i numeri `itemspc_id="@itemspc_id_107"`, `baseitem_id="@baseitem_id_102"`, `partnumber="T0000107"` variano per creare un nuovo articolo specificato. Ad esempio, un nuovo articolo specificato conterrà le voci `itemspc_id="@itemspc_id_108"`, `baseitem_id="@baseitem_id_102"` e `partnumber="T0000108"`, mentre un altro articolo specificato conterrà le voci `itemspc_id`, `baseitem_id` e `partnumber` e così via.

- Utilizzando come guida l'esempio di seguito riportato del negozio ToolTech, aggiungere al database le seguenti informazioni per una relazione tra la versione di un articolo e un articolo di base. Completare questa attività per ciascuna relazione del catalogo:

```
<itemversn
itemversn_id="@itemversn_id_102"
baseitem_id="@baseitem_id_102"
expirationdate="2010-01-01 00:00:00.000000"
versionname="version"
/>
```

dove

- `itemversn_id` è un numero di riferimento generato che identifica la versione dell'articolo.
- `baseitem_id` è l'articolo di base.
- `expirationdate` è l'ora di scadenza delle versione dell'articolo.
- `versionname` identifica in modo univoco la versione dell'articolo per il relativo articolo di base.

Nota: Ogni volta che si crea una relazione tra la versione di un articolo e un articolo di base, i numeri `itemversn_id` e `baseitem_id` variano per creare una nuova relazione. `baseitem_id` corrisponde a un articolo di base esistente. Ad esempio, una nuova relazione, conterrà le voci `itemversn_id="@itemversn_id_107"` e `baseitem_id="@baseitem_id_107"`, mentre un'altra relazione conterrà le voci `itemversn_id="@itemversn_id_108"` e `baseitem_id="@baseitem_id_108"` e così via.

- Utilizzando come guida l'esempio di seguito riportato del negozio ToolTech, aggiungere al database le seguenti informazioni per una relazione tra la versione di un prodotto e un articolo specificato. Completare questa attività per ciascuna relazione del catalogo:

```
<versionspc
versionspc_id="@versionspc_id_106"
itemspc_id="@itemspc_id_106"
itemversn_id="@itemversn_id_102"
/>
```

dove

- `versionspc_id` è l'identificativo univoco generato.
- `itemspc_id` è l'articolo specificato a cui è collegata la voce di catalogo.
- `itemversn_id` identifica la versione dell'articolo.

Nota: Ogni volta che si crea una relazione tra la versione di un prodotto e un articolo specificato, i numeri `versionspc_id` e `itemspc_id` variano per creare una nuova relazione. `itemspc_id` corrisponde a un articolo specificato esistente. Ad esempio, una nuova relazione conterrà le voci

versionspc_id="@versionspc_id_107" e itemspc_id="@itemspc_id_107",
mentre un'altra relazione conterrà le voci
versionspc_id="@versionspc_id_108" e itemspc_id="@itemspc_id_108" e
così via.

5. Utilizzando come guida l'esempio di seguito riportato del negozio ToolTech, aggiungere disposizioni di distribuzione al database. Una disposizione di distribuzione consente a un negozio di vendere la propria merce. Completare questa attività per ciascuna disposizione di distribuzione presente nel catalogo:

```
<distarrang  
distarrang_id="@distarrang_id_102"  
wholesalestore_id="@storeent_id_1"  
merchantstore_id="@storeent_id_1"  
baseitem_id="@baseitem_id_102"  
pickingmethod="F"  
startdate="2000-12-25 00:00:00.000000"  
enddate="2010-01-01 00:00:00.000000"  
>
```

dove

- distarrang_id è il numero di riferimento della disposizione di distribuzione.
- wholesalestore_id è il negozio all'ingrosso a cui appartiene la merce che può essere venduta dal negozio del venditore. Il negozio all'ingrosso deve corrispondere a merchantstore_id.
- merchantstore_id è il negozio del venditore che può vendere la merce del negozio all'ingrosso. Questo negozio del venditore deve corrispondere a wholesalestore_id.
- baseitem_id è il prodotto coperto dalla disposizione di distribuzione.
- pickingmethod determina la sequenza in cui la merce viene scelta dalla tabella RECEIPT in base a questa disposizione:
 - F = FIFO (First In First Out) - la merce ricevuta meno recentemente.
 - L = LIFO (Last in First Out) - la merce ricevuta più recentemente.
- startdate è l'ora in cui questa disposizione di distribuzione diventa effettiva.
- enddate è l'ora in cui questa disposizione di distribuzione non è più effettiva.

Nota: Ogni volta che si crea una disposizione di distribuzione, i numeri distarrang_id e baseitem_id variano per creare una nuova disposizione di distribuzione. Ad esempio, una seconda disposizione di distribuzione conterrà i valori distarrang_id="@distarrang_id_147" e baseitem_id="@baseitem_id_147", mentre una terza disposizione conterrà i valori distarrang_id="@distarrang_id_192" e baseitem_id="@baseitem_id_192" e così via.

6. Utilizzando come guida l'esempio di seguito riportato del negozio ToolTech, aggiungere al database gli attributi che determinano il modo in cui un particolare negozio assegna l'inventario per gli articoli specificati di un particolare articolo di base. Completare questa attività per ciascun articolo di base presente nel catalogo:

```
<storeitem  
baseitem_id="@baseitem_id_102"  
storeent_id="@storeent_id_1"  
trackinventory="Y"  
forcebackorder="N"  
releaseseparately="N"  
returnnotdesired="N"
```

```
backorderable="Y"  
creditable="Y"  
minqtyforsplit="0"  
</>
```

dove

- `baseitem_id` è l'articolo di base.
- `storeent_id` è il negozio o il gruppo di negozi.
- `trackinventory` controlla se l'inventario è registrato nella tabella RECEIPT:
 - N = L'inventario non è registrato e non ci sono voci nella tabella RECEIPT.
 - Y = L'inventario è registrato nella tabella RECEIPT.
- `forcebackorder` viene temporaneamente sospesa l'allocazione degli articoli specificati per l'articolo di base:
 - N = L'inventario può essere allocato (comportamento normale).
 - Y = L'inventario non può essere allocato, sebbene la merce in magazzino sia sufficiente.
- `releaseseparately` controlla il modo in cui vengono rilasciati gli articoli dell'ordine specificati per l'articolo di base:
 - N = Gli articoli dell'ordine possono essere rilasciati con gli altri articoli dell'ordine.
 - Y = Gli articoli dell'ordine devono essere rilasciati separatamente (nelle relative caselle).
- `returnnotdesired` indica che non è consentita la restituzione di un articolo (ad esempio per alimenti deperibili), anche se il cliente ha intenzione o desidera restituire l'articolo:
 - N = Richiesta di credito valutata in base all'intenzione dei clienti di restituire l'articolo, ma la restituzione non è prevista.
 - Y = Richiesta di credito valutata come se la restituzione fosse prevista.
- `backorderable` indica che gli articoli specificati per l'articolo di base non possono essere messi in backorder:
 - N = Gli articoli non possono essere messi in backorder.
 - Y = Gli articoli possono essere messi in backorder.
- `creditable` indica se il venditore, senza annullamento, emetterà un credito per questo articolo:
 - N = Venduto as-is.
 - Y = Accredicabile.
- `minqtyforsplit` indica che gli articoli dell'ordine non verranno automaticamente divisi durante l'allocazione dell'inventario se la quantità rimanente non allocata nel nuovo articolo dell'ordine è inferiore alla quantità minima specificata.

Nota: Ogni volta che si definiscono regole di allocazione dell'inventario per l'articolo di un negozio, il numero `baseitem_id` varia per rappresentare un nuovo articolo di base. Ad esempio, una nuova allocazione conterrà il valore `baseitem_id="@baseitem_id_147"` mentre una terza allocazione conterrà il valore `baseitem_id="@baseitem_id_192"` e così via.

7. Utilizzando come guida l'esempio di seguito riportato del negozio ToolTech, aggiungere la descrizione dell'articolo di base al file XML specifico delle locale a scopi di conversione. Completare questa attività per ciascuna descrizione di articolo di base presente nel catalogo:

```

<baseitmdsc
baseitem_id="@baseitem_id_102"
language_id="&en_US;"
shortdescription="Circular Saw"
longdescription="Light on weight but not in quality. The Circular Saw
weighs a maximum of 10.9lbs., with a choice of a 12 or 14 amp motor,
and speeds of up to 600 rpms! Low friction 220V aluminum alloy shoe
will ensure the job gets done on time."
/>

```

dove

- baseitem_id è la chiave univoca generata.
- language_id è la lingua di queste informazioni.
- shortdescription è una descrizione breve dell'articolo di base.
- longdescription è una descrizione dettagliata dell'articolo di base.

Parte 5: Creazione delle voci di catalogo

1. Utilizzando come guida l'esempio di seguito riportato del negozio ToolTech, creare voci di catalogo aggiungendo informazioni alle tabelle CATENTRY e CATENTDESC. Ciascun tipo di voce di catalogo — prodotti, articoli, pacchetti, bundle e kit dinamici — rappresenta un tipo di merce che è possibile ordinare dal catalogo. E' necessario definire un articolo di base per ciascuna voce di catalogo del prodotto. Completare questa attività per ciascuna voce di catalogo del prodotto presente nel catalogo:

```

<catentry
catentry_id="@product_id_102"
baseitem_id="@baseitem_id_102"
member_id="&MEMBER_ID"
catenttype_id="ProductBean"
partnumber="T0000102"
mfpartnumber="Sprain-Tools-102"
mfname="Sprain Tools"
markfordelete="0"
buyable="1"
/>

```

dove

- catentry_id è il numero di riferimento interno della voce di catalogo del prodotto.
- baseitem_id è l'articolo di base a cui è collegata la voce di catalogo.
- member_id è il numero di riferimento che identifica la voce di catalogo.
- catenttype_id identifica il tipo di voce di catalogo:
 - ItemBean = identifica un articolo.
 - ProductBean = identifica un prodotto.
 - PackageBean = identifica un pacco.
 - BundleBean = identifica un bundle.
 - DynamicKitBean = identifica un kit dinamico.
- partnumber è il numero di riferimento che identifica il numero di parte della voce di catalogo.
- mfpartnumber è il numero di parte utilizzato dal produttore per identificare la voce di catalogo.
- mfname è il nome del produttore della voce di catalogo.
- markfordelete indica se la voce di catalogo specificata è stata contrassegnata per la cancellazione:

- 0 = No.
- 1 = Sì.
- buyable indica se è possibile acquistare la voce di catalogo separatamente:
 - 0 = No.
 - 1 = Sì.

Nota: Ogni volta che si aggiunge una voce di catalogo, le sequenze di `catentry_id` e `baseitem_id` variano per rappresentare una nuova voce di catalogo. `catenttype_id` varia in base al tipo di voce di catalogo.

- Utilizzando come guida l'esempio di seguito riportato del negozio ToolTech, definire un articolo specificato per ciascuna voce di catalogo. Completare questa attività per ciascuna voce presente nel catalogo:

```
<catentry
catentry_id="@catentry_id_106"
itemspc_id="@itemspc_id_106"
member_id="MEMBER_ID"
catenttype_id="ItemBean"
partnumber="T0000106"
mfpartnumber="Sprain-Tools-106"
mfname="Sprain Tools"
markfordelete="0"
buyable="1"
/>
```

dove

- `catentry_id` è il numero di riferimento interno della voce di catalogo.
- `itemspc_id` è l'articolo specificato a cui appartiene la voce di catalogo.
- `member_id` è il numero di riferimento che identifica la voce di catalogo.
- `catenttype_id` identifica il tipo di voce di catalogo:
 - ItemBean = identifica un articolo.
 - ProductBean = identifica un prodotto.
 - PackageBean = identifica un pacco.
 - BundleBean = identifica un bundle.
 - DynamicKitBean = identifica un kit dinamico.
- `partnumber` è il numero di riferimento che identifica il numero di parte della voce di catalogo.
- `mfpartnumber` è il numero di parte utilizzato dal produttore per identificare la voce di catalogo.
- `mfname` è il nome del produttore della voce di catalogo.
- `markfordelete` indica se la voce di catalogo specificata è stata contrassegnata per la cancellazione:
 - 0 = No.
 - 1 = Sì.
- buyable indica se è possibile acquistare la voce di catalogo separatamente:
 - 0 = No.
 - 1 = Sì.

Nota: Ogni volta che si aggiunge un articolo specificato a una voce di catalogo, le sequenze di `catentry_id` e `itemspc_id` variano per rappresentare una nuova voce di catalogo. `catenttype_id` varia in base al tipo di voce di catalogo. Per le limitazioni strutturali del catalogo

principale, una voce di catalogo non può appartenere a più categorie. Per inserire una voce di catalogo in più di una categoria, utilizzare il catalogo di navigazione.

- Utilizzando come guida l'esempio di seguito riportato del negozio ToolTech, aggiungere la descrizione nel file XML specifico delle locale. Completare questa attività per ciascuna descrizione di voce presente nel catalogo:

```
<catentdesc
catentry_id="@product_id_102"
language_id="&en_US"
name="Circular"
shortdescription="Circular Saw"
longdescription="Light on weight but not in quality. The Circular Saw
weighs a maximum of 10.9lbs., with a choice of a 12 or 14 amp motor,
and speeds of up to 600 rpms! Low friction 220V aluminum alloy shoe
will ensure the job gets done on time."
thumbnail="images/circular_saw_sm.gif"
fullimage="images/circular_saw.gif"
available="1"
published="1"
/>
```

dove

- `catentry_id` è il numero di riferimento interno che indica la voce di catalogo a cui si riferiscono queste informazioni specifiche della lingua.
- `language_id` è l'identificativo della lingua.
- `name` è il nome che dipende dalla lingua della voce di catalogo.
- `shortdescription` è una descrizione breve della voce di catalogo.
- `longdescription` è una descrizione dettagliata della voce di catalogo.
- `thumbnail` è il percorso per l'immagine in miniatura.
- `fullimage` è il percorso per l'immagine completa.
- `available` indica per quanto tempo è disponibile la voce di catalogo.
- `published` indica se questa voce di catalogo deve essere visualizzata per la lingua indicata da `language_id`:
 - 0 = Display.
 - 1 = Not display.

Parte 6: Creazione degli attributi e dei valori di attributo

1. Utilizzando come guida l'esempio di seguito riportato del negozio ToolTech, creare attributi e valori di attributi per i prodotti, aggiungendo informazioni alle tabelle ATTRIBUTE e ATTRVALUE nel file XML specifico delle locale a scopo di conversione. Ogni prodotto del catalogo ha un determinato insieme di attributi, ad esempio la taglia ed il colore per una camicia o un paio di pantaloni. Gli articoli sono definiti in base ai valori degli attributi. Ad esempio, mentre una camicia rappresenta un prodotto, una camicia media nera rappresenta un articolo. Completare questa attività per ciascun attributo presente nel catalogo:

```
<attribute
attribute_id="@attribute_id_103"
language_id="&en_US"
attrtype_id="STRING"
name="Amps"
sequence="0"
description="Amps"
catentry_id="@product_id_102"
description2="Amps"
/>
```

dove

- attribute_id è il numero di riferimento interno dell'attributo.
- language_id è la lingua dell'attributo.
- attrtype_id è il tipo di valore corrispondente dell'attributo.
- name è il nome dell'attributo.
- sequence è un numero in sequenza che determina l'ordine di visualizzazione degli attributi per un determinato prodotto.
- description è la descrizione dell'attributo.
- catentry_id è il numero di riferimento del prodotto a cui l'attributo appartiene.
- description2 è una descrizione supplementare dell'attributo.

Nota: Ogni volta che si aggiunge un attributo a un prodotto, definito da catentry_id, la sequenza di attribute_id varia per rappresentare un nuovo attributo.

2. Utilizzando come guida l'esempio di seguito riportato del negozio ToolTech, aggiungere i valori dell'attributo. Completare questa attività per ciascun valore di attributo presente nel catalogo:

```
<attrvalue  
attrvalue_id="@attrvalue_id_114"  
language_id="&en_US"  
attribute_id="@attribute_id_103"  
name="12.0&amps"  
attrtype_id="STRING"  
stringvalue="12.0&amps"  
sequence="0"  
catentry_id="@catentry_id_106"  
>
```

dove

- attrvalue_id è il numero di riferimento interno dell'attributo.
- language_id è la lingua del valore dell'attributo.
- attribute_id è il numero di riferimento interno dell'attributo associato al valore.
- name è il nome del valore dell'attributo.
- attrtype_id è il tipo di valore dell'attributo.
- stringvalue è il valore dell'attributo.
- sequence è un numero in sequenza che determina l'ordine di visualizzazione dei valori degli attributi per un determinato attributo.
- catentry_id è l'ID dell'articolo descritto dal valore di questo attributo.

Nota: Ogni volta che si aggiunge un valore di attributo a un attributo, la sequenza di attrvalue_id varia per rappresentare valori diversi. La sequenza di attribute_id varia per rappresentare un attributo diverso. Il valore di sequence aumenta con ogni nuovo valore di attributo. Ad esempio, i valori di attributo successivi saranno sequence="1", sequence="2", sequence="3" e così via.

Parte 7: Creazione delle relazioni tra prodotti e articoli

1. Una volta creati prodotti e articoli del catalogo, definire le relazioni tra i prodotti e gli articoli aggiungendo informazioni alla tabella CATENTREL.

Utilizzare come guida l'esempio di seguito riportato del negozio ToolTech. Completare questa attività per ciascuna relazione tra prodotto e articolo presente nel catalogo:

```
<catentrel
catentry_id_parent="@product_id_147"
catreltype_id="PRODUCT_ITEM"
catentry_id_child="@catentry_id_152"
sequence="2"
quantity="1"
/>
```

dove

- `catentry_id_parent` è il numero di riferimento della voce di catalogo di origine nella relazione, vale a dire il prodotto.
- `catreltype_id` è il tipo di relazione: `PRODUCT_ITEM`
- `catentry_id_child` è il numero di riferimento della voce di catalogo di destinazione nella relazione, vale a dire l'articolo.
- `sequence` è il numero in sequenza utilizzato per determinare l'ordine di visualizzazione.
- `quantity` è una quantità che può essere associata alla relazione.

Nota: Ogni volta che si aggiunge una relazione tra un prodotto e un articolo, i numeri `catentry_id_parent` e `catentry_id_child` variano per creare relazioni diverse, in base al valore di `catreltype_id`. Con ciascuna nuova relazione, il numero di `sequence` è diverso. Ad esempio, con `sequence="2"`, la relazione successiva sarà `sequence="3"`, seguita da `sequence="4"` e così via.

Parte 8: Creazione di pacchetti e bundle

1. Una volta creati prodotti e articoli, creare pacchetti e bundle aggiungendo informazioni alle tabelle `CATENTRY`, `CATENTDESC` e `CATENTREL`. Utilizzando come guida l'esempio di seguito riportato del negozio ToolTech, iniziare a creare un pacchetto o un bundle aggiungendo informazioni alla tabella `CATENTRY`. Completare questa attività per ciascun pacchetto e Bundle presente nel catalogo:

```
<catentry
catentry_id="@package_id_102"
member_id="@MEMBER_ID"
catenttype_id="PackageBean"
partnumber="sku-@package_id_102"
mfpartnumber="sku-@package_id_102"
mfname="ToolTech"
markfordelete="0"
buyable="1"
/>
```

dove

- `catentry_id` è il numero di riferimento della voce di catalogo.
- `member_id` è il numero di riferimento che identifica il proprietario del catalogo.
- `catenttype_id` identifica il tipo di voce di catalogo:
 - `PackageBean` = identifica un pacco.
 - `BundleBean` = identifica un bundle.
- `partnumber` è il numero di riferimento che identifica il numero di parte della voce di catalogo.

- mfpnumber è il numero di parte utilizzato dal produttore per identificare la voce di catalogo.
- mfname è il nome del produttore della voce di catalogo.
- markfordelete indica se la voce di catalogo è stata contrassegnata per la cancellazione:
 - 0 = No.
 - 1 = Sì.
- buyable indica se la voce di catalogo può essere acquistata singolarmente:
 - 0 = No.
 - 1 = Sì.

Nota: Ogni volta che si crea un pacchetto o un bundle, i numeri catentry_id, partnumber e mfpnumber variano per creare pacchetti o bundle diversi. Ad esempio, per creare un nuovo pacchetto, è possibile utilizzare i valori catentry_id="@package_id_103", partnumber="sku-@package_id_103" e mfpnumber="sku-@package_id_103", incluso catenttype_id="PackageBean" per identificare la voce come un pacchetto. Per creare un nuovo bundle, è possibile utilizzare i valori catentry_id="@package_id_110", partnumber="sku-@package_id_110" e mfpnumber="sku-@package_id_110", incluso catenttype_id="BundleBean" per identificare la voce come un bundle e così via.

- Utilizzando come guida l'esempio di seguito riportato del negozio ToolTech, inserire la descrizione del pacchetto o bundle aggiungendo informazioni alla tabella CATENTDESC nel file XML specifico delle locale a scopi di conversione. Completare questa attività per ciascuna descrizione di pacchetto e bundle presente nel catalogo:

```
<catentdesc
catentry_id="@catentry_id_102"
language_id="-1"
name="computer"
shortdescription="Computer"
longdescription="A combination of a central processing unit, monitor,
hard drive, and color printer. An ideal starter system."
thumbnail="images/package_system_sm.gif"
fullimage="images/package_system.gif"
available="1"
published="1"
/>
```

dove

- catentry_id è il numero di riferimento interno che indica la voce di catalogo a cui si riferiscono queste informazioni specifiche della lingua.
- language_id è l'identificativo della lingua.
- name è il nome che dipende dalla lingua della voce di catalogo.
- shortdescription è una descrizione breve della voce di catalogo.
- longdescription è una descrizione dettagliata della voce di catalogo.
- thumbnail è il percorso dell'immagine in miniatura per la voce di catalogo.
- fullimage è il percorso dell'immagine completa per la voce di catalogo.
- available indica per quanto tempo è disponibile la voce di catalogo.
- published indica se la voce di catalogo deve essere visualizzata per la lingua indicata da language_id:
 - 0=Non visualizzare la voce di catalogo.

- 1=Visualizzare la voce di catalogo.
- Utilizzando come guida l'esempio di seguito riportato del negozio ToolTech, creare relazioni tra pacchetti o bundle e relativi componenti aggiungendo informazioni alla tabella CATENTREL. Completare questa attività per ciascuna relazione tra i componenti dei pacchetti o bundle presenti nel catalogo:

```
<catentrel
catentry_id_parent="@catentry_id_102"
catreltype_id="PACKAGE_COMPONENT"
catentry_id_child="@catentry_id_97"
sequence="1.0"
quantity="1.0"
/>
```

dove

- catentry_id_parent è il numero di riferimento della voce di catalogo di origine nella relazione, vale a dire il pacchetto o il bundle.
- catreltype_id è il tipo di relazione:
 - PACKAGE_COMPONENT rappresenta una relazione tra un pacchetto e i relativi componenti.
 - BUNDLE_COMPONENT rappresenta una relazione tra un bundle e i relativi componenti.
- catentry_id_child è il numero di riferimento della voce di catalogo di destinazione nella relazione, vale a dire il componente.
- sequence è il numero in sequenza utilizzato per determinare l'ordine di visualizzazione.
- quantity è una quantità che può essere associata alla relazione.

Nota: Ogni volta che si crea una relazione tra un pacchetto e un bundle, i numeri catentry_id_parent e catentry_id_child variano per creare una corrispondenza con le voci di catalogo esistenti. Con ciascuna nuova relazione, il numero di sequence è diverso. Ad esempio, se si inizia con sequence="1.0", la relazione successiva sarà sequence="2.0", seguita da sequence="3.0" e così via.

Parte 9: Creazione delle relazioni tra gruppi di catalogo e voci di catalogo

1. Una volta creati gruppi di catalogo e voci di catalogo, definirne le relazioni aggiungendo informazioni alla tabella CATGPENREL. Per le limitazioni strutturali del catalogo principale, una voce di catalogo non può appartenere a più categorie. Per inserire una voce di catalogo in più di una categoria, utilizzare il catalogo di navigazione. Utilizzare come guida l'esempio di seguito riportato del negozio ToolTech. Completare questa attività per ciascuna relazione tra gruppi di catalogo e voci di catalogo presente nel catalogo:

```
<catgpenrel
catgroup_id="@catgroup_id_11"
catalog_id="@catalog_id_1"
catentry_id="@product_id_102"
sequence="0"
/>
```

dove

- catgroup_id è il gruppo di catalogo di origine di questa relazione.
- catalog_id è il catalogo in cui è stata rilevata questa relazione.
- catentry_id è la voce di catalogo di destinazione di questa relazione.

- sequence è il numero di sequenza che determina l'ordine di visualizzazione del contenuto del gruppo di catalogo.

Nota: Ogni volta che si crea una relazione tra gruppi di catalogo e voci di catalogo, i numeri catgroup_id e catentry_id variano per formare nuove relazioni con gruppi di catalogo e voci di catalogo diversi. Con ciascuna nuova relazione, il numero di sequence è diverso. Ad esempio, se si inizia con sequence="0", la relazione successiva sarà sequence="1", seguita da sequence="2" e così via.

Parte 10: Creazione delle associazioni commerciali

1. Utilizzando come guida l'esempio di seguito riportato del negozio ToolTech, creare associazioni commerciali tra le voci di catalogo aggiungendo informazioni alla tabella MASSOCECE. Completare questa attività per ciascuna associazione commerciale presente nel catalogo:

```
<massoccece
massoccece_id="@relationship_id_100"
massoctype_id="X-SELL"
catentry_id_from="@product_id_1"
catentry_id_to="@product_id_15"
massoc_id="REQUIRES"
quantity="2.0"
rank="1.00000"
/>
```

dove

- massoccece_id è il numero di riferimento di questa voce.
- massoctype_id è l'identificativo del tipo di associazione:
 - X-SELL = cross-sell.
 - UPSELL = up-sell.
 - ACCESSORY = accessory.
- catentry_id_from è la voce di catalogo che costituisce l'origine dell'associazione.
- catentry_id_to è la voce di catalogo che costituisce la destinazione dell'associazione.
- massoc_id è l'identificativo dello specificatore semantico:
 - REQUIRED
 - OPTIONAL
 - COMES WITH
- quantity è la quantità relativa a questa associazione.
- rank è il numero in sequenza utilizzato per l'ordine di visualizzazione.

Nota: Ogni volta che si aggiunge un'associazione commerciale, il numero massoccece_id varia per rappresentare una nuova relazione. I numeri catentry_id_from e catentry_id_to variano per creare un nuovo contenuto commerciale per l'associazione.

Parte 11: Associazione del catalogo a un negozio

1. Associare il catalogo ad un negozio, assegnando il catalogo, i relativi gruppi di catalogo e le voci di catalogo ad un negozio del database utilizzando come guida il file storecatalog.xml esistente del negozio di esempio ToolTech. E' necessario anche assegnare pagine di visualizzazione ai gruppi di catalogo e alle voci di catalogo. Aggiungere queste informazioni alle tabelle STORECAT,

STORECENT, STORECGRP, DISPCGPREL e DISPENTREL. Se il catalogo da creare è multiculturale, creare un file XML di relazioni negozio-catalogo diverso per ogni locale supportata dal negozio.

```
<storecat
catalog_id="@catalog_id_1"
storeent_id="@storeent_id_1"
mastercatalog="1"
/>
```

dove

- catalog_id è il numero di riferimento del catalogo.
- storeent_id è il numero di riferimento dell'entità negozio del database.
- mastercatalog specifica un catalogo principale per il negozio. Un valore pari a 1 indica che questo catalogo viene indicato come catalogo principale.

2. Utilizzando come guida l'esempio di seguito riportato del negozio ToolTech, aggiungere voci di catalogo alla relazione negozio-catalogo. Completare questa attività per ciascuna voce presente nel catalogo:

```
<storecent
storeent_id="@storeent_id_1"
catentry_id="@product_id_102"
/>
```

dove

- storeent_id è il numero di riferimento dell'entità negozio del database.
- catentry_id è il numero di riferimento della voce di catalogo.

Nota: Ogni volta che si aggiunge un numero catentry_id all'entità negozio, il numero di riferimento varia per creare una corrispondenza con una voce di catalogo esistente.

3. Utilizzando come guida l'esempio di seguito riportato del negozio ToolTech, aggiungere gruppi di catalogo all'entità negozio. Completare questa attività per ciascun gruppo di catalogo presente nel catalogo:

```
<storecgrp
storeent_id="@storeent_id_1"
catgroup_id="@catgroup_id_1"
/>
```

dove

- storeent_id è il numero di riferimento dell'entità negozio del database.
- catgroup_id è il numero di riferimento del gruppo di catalogo.

Nota: Ogni volta che si aggiunge un numero catgroup_id all'entità negozio, il numero di riferimento varia per creare una corrispondenza con un gruppo di catalogo esistente.

Parte 12: Associazione delle tasse al catalogo

1. Associare tasse a prodotti e servizi del catalogo di un determinato negozio. E' necessario associare un codice di calcolo delle tasse alle voci di catalogo aggiungendo queste informazioni alla tabella CATENCALCD. Per ulteriori informazioni, consultare la sezione "Creazione di risorse tasse in WebSphere Commerce" a pagina 156.

Parte 13: Associazione dei metodi di spedizione al catalogo

1. Per assegnare metodi di spedizione a prodotti e servizi del catalogo, è necessario associare un codice di calcolo per la spedizione alle voci di catalogo. Aggiungere queste informazioni alla tabella CATENCALCD. Per ulteriori informazioni, consultare la sezione "Creazione di risorse spedizione in WebSphere Commerce" a pagina 137.

Parte 14: Associazione di un centro di evasione ordini al catalogo

1. Associare il catalogo a un centro di evasione per spedire i prodotti ai clienti. Un centro di evasione ordini gestisce l'inventario dei prodotti e la spedizione per un negozio. Aggiungere queste informazioni alla tabella FFMCENTER. Per ulteriori informazioni, consultare la sezione "Creazione di risorse evasione ordini in WebSphere Commerce" a pagina 112.

Parte 15: Creazione dei prezzi per le voci di catalogo

1. Creare l'assegnazione prezzi per le voci di catalogo. L'assegnazione prezzi rappresenta l'intervallo di prezzi per una voce di catalogo ed i criteri che devono essere soddisfatti per l'uso di quel prezzo. Per creare un catalogo funzionale, è necessario aggiungere informazioni sull'offerta al database. Aggiungere queste informazioni alle tabelle TRADEPOSCN, TDPSCNCNTR, MGPTRDPSCN, OFFER e OFFERPRICE. Per ulteriori informazioni, consultare la sezione "Creazione di risorse assegnazione prezzi in WebSphere Commerce" a pagina 92. In alternativa, è possibile creare o aggiornare l'assegnazione prezzi di una voce di catalogo mediante gli strumenti di gestione del prodotto di WebSphere Commerce Accelerator.

Parte 16: Caricamento del file XML

1. Una volta creati i dati, caricare il file XML nel database utilizzando il pacchetto Loader o la funzione di pubblicazione di Servizi del negozio. Per ulteriori informazioni sul pacchetto Loader, consultare la Parte 7, "Pubblicazione del negozio" a pagina 213.

Nota: E' possibile utilizzare anche gli strumenti di gestione dei prodotti di WebSphere Commerce Accelerator per creare risorse catalogo per il catalogo principale. Per informazioni più dettagliate sugli strumenti di gestione dei prodotti, fare riferimento alla guida in linea di WebSphere Commerce.

Visualizzazione delle risorse catalogo del negozio

Una volta associato un catalogo, gruppi di catalogo e voci di catalogo ad un negozio, assegnare modelli JSP per la visualizzazione delle voci di catalogo e dei gruppi di catalogo creando queste relazioni nel database. Creare queste relazioni in formato file XML che possono essere caricati nel database mediante il pacchetto Loader.

Il file storecatalog.xml del negozio ToolTech si trova nel file dell'archivio negozio. Per visualizzare il file storecatalog.xml, decomprimere l'archivio negozio mediante un programma ZIP. Il file storecatalog.xml si trova nelle seguenti directory di dati:

-  *unità:*:\WebSphere\CommerceServer\samplstores
-  *unità:*:\Program Files\WebSphere\CommerceServer\samplstores
-  /usr/WebSphere/CommerceServer/samplstores

-   /opt/WebSphere/CommerceServer/samplestores
-  /qibm/proddata/WebCommerce/samplestores

Nota: La guida in linea di WebSphere Commerce contiene le informazioni su ciascuna delle risorse dati contenute nei negozi di esempio.

Il file store-catalog.dtd si trova nella seguente directory:

-  *unità*: \WebSphere\CommerceServer\xml\sar
-  *unità*: \Program Files\WebSphere\CommerceServer\xml\sar
-  /usr/WebSphere/CommerceServer/xml/sar
-   /opt/WebSphere/CommerceServer/xml/sar
-  /qibm/proddata/WebCommerce/xml/sar

Prima di creare le relazioni negozio-catalogo, verificare che le risorse dati del negozio siano state create. Completare le seguenti attività, ognuna delle quali crea voci nel file storecatalog.xml:

1. Per visualizzare i gruppi di catalogo (categorie) nel negozio, è necessario assegnare modelli JSP ai gruppi di catalogo. E' possibile assegnare un determinato modello di pagina di visualizzazione ad un gruppo di catalogo o un modello predefinito che visualizzi tutti i gruppi di catalogo. Utilizzando come guida l'esempio di seguito riportato del negozio ToolTech, assegnare i modelli dei gruppi di catalogo aggiungendo informazioni alla tabella DISPCGPREL. Completare questa attività per ciascun modello che si desidera assegnare ai gruppi di catalogo.

```
<dispcgprel
catgroup_id="@catgroup_id_1"
devicefmt_id="-1"
dispcgprel_id="@dispcgprel_id_1"
mbrgrp_id="0"
pagename="/ToolTech/CategoryDisplay.jsp"
storeent_id="@storeent_id_1"
rank="0"/>
```

dove

- catgroup_id è il numero di riferimento del gruppo di catalogo per cui sarà visualizzato questo nome di pagina. Un valore pari a 0 indica che questo nome di pagina verrà utilizzato per tutti i gruppi di catalogo.
- devicefmt_id è il numero di riferimento del tipo di unità su cui la pagina verrà visualizzata. Un valore pari a -1 indica che questa pagina del modello verrà utilizzata da un browser HTTP.
- dispcgprel_id è il numero di riferimento di questa voce.
- mbrgrp_id è il numero di riferimento del gruppo di membri per cui sarà visualizzata questa pagina del modello. Un valore pari a 0 indica che questa pagina del modello verrà utilizzata per tutti i gruppi di membri.
- pagename rappresenta il nome ed il percorso della pagina del modello di visualizzazione.
- rank è un numero di sequenza utilizzato per interrompere i collegamenti quando più pagine soddisfano i criteri di selezione.

Nota: Ogni volta che si assegna un modello JSP ad un gruppo di catalogo, la sequenza di catentry_id varia per creare una corrispondenza con una voce di catalogo esistente.

2. Per visualizzare le voci di catalogo (prodotti, articoli, pacchetti, bundle e kit dinamici) nel negozio, è necessario assegnare modelli JSP alle voci di catalogo. E' possibile assegnare un modello predefinito che visualizzi tutte le voci di catalogo oppure un modello predefinito che visualizzi ciascun tipo di voce di catalogo, ad esempio, un modello per i prodotti ed un altro modello per gli articoli, oppure uno specifico modello per ogni specifica voce di catalogo. Utilizzando come guida l'esempio di seguito riportato del negozio ToolTech, assegnare i modelli aggiungendo informazioni alla tabella DISPENTREL. Completare questa attività per ciascun modello che si desidera assegnare alle voci di catalogo.

```
<dispentrel  
auctionstate="0"  
catentry_id="0"  
catenttype_id="ProductBean"  
devicefmt_id="-1"  
dispentrel_id="@dispentrel_id_1"  
mbrgrp="0"  
pagename="/ToolTech/ProductDisplay.jsp"  
storeent_id="@storeent_id_1"  
rank="0"/>
```

dove

- auctionstate indica che questa pagina del modello visualizza una voce di catalogo messa all'asta:
 - 0=Modello non all'asta.
 - 1=Modello all'asta.
- catentry_id è il numero di riferimento della voce di catalogo per cui sarà visualizzato questo nome di pagina. Un valore pari a 0 indica che questo nome di pagina verrà utilizzato per tutte le voci di catalogo.
- catenttype_id è il tipo di voce di catalogo per visualizzare la quale sarà utilizzata questa pagina:
 - ProductBean=Visualizza un prodotto.
 - ItemBean=Visualizza un articolo.
 - PackageBean=Visualizza un pacchetto.
 - BundleBean=Visualizza un bundle.
 - DynamicKitBean=Visualizza un kit dinamico.
- devicefmt_id è il numero di riferimento del tipo di unità su cui la pagina verrà visualizzata. Un valore pari a -1 indica che questa pagina del modello verrà utilizzata da un browser HTTP.
- dispentrel_id è il numero di riferimento della voce di catalogo.
- mbrgrp è il numero di riferimento del gruppo di membri per cui sarà visualizzata questa pagina del modello. Un valore pari a 0 indica che questa pagina del modello verrà utilizzata per tutti i gruppi di membri.
- pagename rappresenta il nome ed il percorso della pagina del modello di visualizzazione.
- storeent_id è il numero di riferimento del negozio per cui viene visualizzata questa pagina.
- rank è un numero di sequenza utilizzato per interrompere i collegamenti quando più pagine soddisfano i criteri di selezione.

Nota: Ogni volta che si assegna un modello JSP ad una voce di catalogo, la sequenza di catentry_id varia per creare una corrispondenza con una voce di catalogo esistente.

Creazione del catalogo di navigazione

Un negozio WebSphere Commerce consente due tipi di cataloghi: quello principale e quello di navigazione. I cataloghi di navigazione non devono rispettare i limiti strutturali imposti ai cataloghi principali. Tali cataloghi forniscono una struttura di visualizzazione dinamica che consente di creare cataloghi di navigazione adatti alle esigenze del negozio.

In particolare, i cataloghi di navigazione non devono rispettare i seguenti limiti imposti ai cataloghi principali:

- Un catalogo principale deve rappresentare una struttura ad albero corretta, ovvero non deve contenere cicli e non può utilizzare la seguente struttura: la categoria parent A ha una sottocategoria B; è importante che B e le relative sottocategorie non siano la categoria parent di A.
- Un prodotto non può appartenere a più di una categoria.

Le seguenti attività creano un catalogo di navigazione modificando il catalogo del negozio di esempio NewFashion. Il catalogo creato in questo modo non può più essere classificato come catalogo principale perché con la procedura di seguito riportata vengono introdotti i cicli di categorie e l'assegnazione di alcuni prodotti a più categorie. Un classico catalogo di navigazione viene creato aggiungendo informazioni alle tabelle di relazioni tra categorie: CATGRPREL, che contiene le relazioni tra sottocategorie, e CATGPENREL, che contiene le relazioni tra categoria e prodotto. Sebbene questi esempi riguardino NewFashion, è possibile applicare la stessa procedura al proprio catalogo principale, apportando le necessarie modifiche.

Creazione dei cicli di categorie

Il catalogo NewFashion contiene quattro categorie principali: **Moda uomo**, **Moda donna**, **Nuovi arrivi** e **Promozione home page**. Questo esempio illustra come copiare la categoria **Nuovi arrivi** in **Moda uomo** e copiare **Promozione home page** in **Moda donna** e rinominarla come **Nuovi arrivi**.

Per trasformare il catalogo principale del negozio di esempio NewFashion in un catalogo di navigazione mediante i cicli di categorie, procedere come segue:

1. Pubblicare l'archivio negozio NewFashion per creare il negozio di esempio NewFashion. NewFashion è disponibile in lingua inglese e in una delle nove lingue nazionali fornite con WebSphere Commerce. Scegliere uno dei file `NewFashion_en_US_locale.sar` per la pubblicazione.
2. Con un editor, aprire il file `catalog.xml`. Il file si trova nella seguente directory di WebSphere Commerce:
 -  `unitâ:\WebSphere\CommerceServer\samplestores\NewFashion\locale\data`
 -  `unitâ:\Program Files\WebSphere\CommerceServer\samplestores\NewFashion\locale\data`
 -  `/usr/WebSphere/CommerceServer/samplestores/NewFashion/locale/data`
 -   `/opt/WebSphere/Commerce/samplestores/NewFashion/locale/data`

- `400` /qibm/proddata/WebCommerce/samplestores/NewFashion/locale/data

3. Individuare la sezione di dati CATGRPTEL nel file catalog.xml. Creare una nuova relazione tra categorie di livello superiore **Moda uomo** e **Nuovi arrivi**. Al momento, entrambe le categorie sono di livello superiore. Per una relazione di navigazione, creare una nuova sezione che collochi **Nuovi arrivi** come sottocategoria di **Moda uomo** conservando al tempo stesso la struttura originale. Nella sezione CATGRPTEL, aggiungere il seguente estratto:

```
<catgrpTEL
catgroup_id_parent="@catgroup_id_11"
catgroup_id_child="@catgroup_id_21"
catalog_id="@catalog_id_1"
sequence="7"
/>
```

dove

- catgroup_id_parent è il numero di riferimento interno al gruppo di catalogo della categoria parent definito dal negozio di esempio NewFashion. In questo esempio, @catgroup_id_11 è la categoria **Moda uomo**.
 - catgroup_id_child è il numero di riferimento interno al gruppo di catalogo della categoria child definito dal negozio di esempio NewFashion. In questo esempio, @catgroup_id_21 è la categoria **Nuovi arrivi**.
 - catalog_id è il numero di riferimento interno del catalogo definito dal negozio di esempio NewFashion.
 - sequence è il numero che determina l'ordine di visualizzazione del contenuto del gruppo di catalogo definito dal negozio di esempio NewFashion. In questo esempio, la categoria **Nuovi arrivi** viene visualizzata per ultima, dopo le prime sei sottocategorie di **Moda uomo**.
4. Ripetere il passo sopra riportato, creando questa volta una nuova relazione tra categorie di livello superiore tra **Moda donna** e **Promozione home page**. Al momento, entrambe le categorie sono di livello superiore. Per una relazione di navigazione, creare una nuova sezione che collochi **Promozione home page** come sottocategoria di **Moda donna** conservando al tempo stesso la struttura originale. Nella sezione CATGRPTEL, aggiungere il seguente estratto:

```
<catgrpTEL
catgroup_id_parent="@catgroup_id_20"
catgroup_id_child="@catgroup_id_22"
catalog_id="@catalog_id_1"
sequence="9"
/>
```

dove

- catgroup_id_parent è il numero di riferimento interno al gruppo di catalogo della categoria parent definito dal negozio di esempio NewFashion. In questo esempio, @catgroup_id_20 è la categoria **Moda donna**.
- catgroup_id_child è il numero di riferimento interno al gruppo di catalogo della categoria child definito dal negozio di esempio NewFashion. In questo esempio, @catgroup_id_22 è la categoria **Promozioni home page**.
- catalog_id è il numero di riferimento interno del catalogo definito dal negozio di esempio NewFashion.
- sequence è il numero che determina l'ordine di visualizzazione del contenuto del gruppo di catalogo definito dal negozio di esempio NewFashion. In questo esempio, la categoria **Promozione home page** viene visualizzata per ultima.

5. Ora che **Promozione home page** è una sottocategoria di **Moda donna**, il nome della categoria non è esatto. Rinominare la categoria in **Nuovi arrivi**, in modo che corrisponda alla sottocategoria di **Moda uomo**.
6. Salvare il file `catalog.xml`.
7. Per visualizzare le modifiche apportate, effettuare una delle seguenti operazioni: pubblicare l'archivio di negozio NewFashion modificato mediante Servizi del negozio oppure caricare il file `catalog.xml` con il pacchetto loader come indicato in "Caricamento dei gruppi di risorse database" a pagina 279.

Aggiunta di un prodotto a una seconda categoria

Questo esempio illustra come copiare i prodotti da una categoria all'altra conservando al tempo stesso la struttura originale. La categoria **Promozione home page** contiene il prodotto **Camicia da notte estiva**, che potrebbe appartenere anche alla sottocategoria **Pigiama** della categoria di livello superiore **Moda donna**. Queste istruzioni illustrano come copiare il prodotto **Camicia da notte estiva** e le relative SKU nella categoria **Pigiama**.

Per trasformare il catalogo principale del negozio di esempio NewFashion in un catalogo di navigazione aggiungendo un prodotto a una seconda categoria, procedere come segue:

1. Pubblicare l'archivio negozio NewFashion per creare il negozio di esempio NewFashion. NewFashion è disponibile in lingua inglese e in una delle nove lingue nazionali fornite con WebSphere Commerce. Scegliere uno dei file `NewFashion_en_US_locale.sar` per la pubblicazione.
2. Con un editor, aprire il file `catalog.xml`. Il file si trova nella seguente directory di WebSphere Commerce:
 -  `unità:\WebSphere\CommerceServer\samplestores\NewFashion\locale\data`
 -  `unità:\Program Files\WebSphere\CommerceServer\samplestores\NewFashion\locale\data`
 -  `/usr/WebSphere/CommerceServer/samplestores/NewFashion/locale/data`
 -   `/opt/WebSphere/CommerceServer/samplestores/NewFashion/locale/data`
 -  `/qibm/proddata/WebCommerce/samplestores/NewFashion/locale/data`
3. Individuare la sezione di dati CATGPENREL nel file `catalog.xml`. Creare una nuova voce di prodotto per **Camicia da notte estiva**, originariamente nella categoria **Promozione home page**. Nella sezione CATGPENREL, aggiungere il seguente estratto per includere il prodotto:

```
<catgpenrel
catgroup_id="@catgroup_id_18"
catalog_id="@catalog_id_1"
catentry_id="@product_id_2692"
sequence="2"
/>
```

dove

- `catgroup_id_` è il numero di riferimento interno al gruppo di catalogo definito dal negozio di esempio NewFashion. In questo esempio, `@catgroup_id_18` è la categoria **Pigiama**.

- `catalog_id` è il numero di riferimento interno del catalogo definito dal negozio di esempio NewFashion.
 - `catentry_id_` è il numero di riferimento interno alla voce di catalogo definito dal negozio di esempio NewFashion. In questo esempio, `@catentry_id_2692` è il prodotto **Camicia da notte estiva**.
 - `sequence` è il numero che determina l'ordine di visualizzazione del contenuto del gruppo di catalogo definito dal negozio di esempio NewFashion. In questo esempio, il prodotto **Camicia da notte estiva** viene visualizzato per ultimo.
4. Dopo aver aggiunto la voce di prodotto **Camicia da notte estiva**, aggiungere le voci di SKU per il prodotto nella sezione `CATGPENREL`, come definite nel negozio di esempio NewFashion. Al momento, il prodotto **Camicia da notte estiva** contiene dieci SKU definite. Nella sezione `CATGPENREL`, aggiungere i seguenti estratti per includere le SKU:

```
<catgpenrel
catgroup_id="@catgroup_id_18"
catalog_id="@catalog_id_1"
catentry_id="@catentry_id_2695"
sequence="2"
/>
```

```
<catgpenrel
catgroup_id="@catgroup_id_18"
catalog_id="@catalog_id_1"
catentry_id="@catentry_id_2696"
sequence="2"
/>
```

```
<catgpenrel
catgroup_id="@catgroup_id_18"
catalog_id="@catalog_id_1"
catentry_id="@catentry_id_2697"
sequence="2"
/>
```

```
<catgpenrel
catgroup_id="@catgroup_id_18"
catalog_id="@catalog_id_1"
catentry_id="@catentry_id_2698"
sequence="2"
/>
```

```
<catgpenrel
catgroup_id="@catgroup_id_18"
catalog_id="@catalog_id_1"
catentry_id="@catentry_id_2699"
sequence="2"
/>
```

```
<catgpenrel
catgroup_id="@catgroup_id_18"
catalog_id="@catalog_id_1"
catentry_id="@catentry_id_2700"
sequence="2"
/>
```

```
<catgpenrel
catgroup_id="@catgroup_id_18"
catalog_id="@catalog_id_1"
catentry_id="@catentry_id_2701"
sequence="2"
/>
```

```
<catgpenrel
catgroup_id="@catgroup_id_18"
catalog_id="@catalog_id_1"
catentry_id="@catentry_id_2702"
sequence="2"
/>
```

```
<catgpenrel
catgroup_id="@catgroup_id_18"
catalog_id="@catalog_id_1"
catentry_id="@catentry_id_2703"
sequence="2"
/>
```

```
<catgpenrel
catgroup_id="@catgroup_id_18"
catalog_id="@catalog_id_1"
catentry_id="@catentry_id_2704"
sequence="2"
/>
```

dove

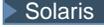
- `catgroup_id` è il numero di riferimento interno al gruppo di catalogo definito dal negozio di esempio NewFashion. In questo esempio, `@catgroup_id_18` è la categoria **Pigiama**.
 - `catalog_id` è il numero di riferimento interno del catalogo definito dal negozio di esempio NewFashion.
 - `catentry_id` è il numero di riferimento interno alla voce di catalogo definito dal negozio di esempio NewFashion. In questo esempio, le SKU da `@catentry_id_2695` a `@catentry_id_2704` rappresentano le dieci SKU definite per il prodotto **Camicia da notte estiva**.
 - `sequence` è il numero che determina l'ordine di visualizzazione del contenuto del gruppo di catalogo definito dal negozio di esempio NewFashion. In questo esempio, le SKU di **Camicia da notte estiva** vengono visualizzate per ultimi.
5. Salvare il file `catalog.xml`.
 6. Per visualizzare le modifiche apportate, effettuare una delle seguenti operazioni: pubblicare l'archivio di negozio NewFashion modificato mediante Servizi del negozio oppure caricare il file `catalog.xml` con il pacchetto loader come indicato in "Caricamento dei gruppi di risorse database" a pagina 279.

Creazione delle risorse catalogo in WebSphere Commerce

Nel tempo, sarà necessario aggiornare le informazioni sulle risorse database del catalogo principale. La gestione del catalogo è un processo di routine, in quanto è necessario aggiungere e rimuovere continuamente merce, creare e associare le categorie o i gruppi di catalogo e aggiornare le informazioni sui prodotti, ad esempio le descrizioni e il prezzo.

E' possibile cambiare le risorse catalogo modificando i dati XML di WebSphere Commerce utilizzando le voci di database esistenti e i file `catalog.xml` del negozio. Utilizzare i file XML del negozio di esempio di WebSphere Commerce come riferimento. Tali file risiedono nella seguente directory di dati:

-  `unità:\WebSphere\CommerceServer\samplstores`
-  `unità:\Program Files\WebSphere\CommerceServer\samplstores`
-  `/usr/WebSphere/CommerceServer/samplstores`

-   /opt/WebSphere/CommerceServer/samplestores
-  /qibm/proddata/WebCommerce/samplestores

Nota: Questi esempi sono tratti dal negozio di esempio NewFashion e identificano gli elementi XML da modificare per cambiare le informazioni sulle risorse catalogo.

Gruppi di catalogo

I gruppi di catalogo vengono creati nel catalogo WebSphere Commerce utilizzando le tabelle di database CATGROUP e CATGRPDESC. Dal file catalog.xml, un tipico gruppo di catalogo appare simile al seguente estratto:

```
<catgroup
  catgroup_id="@catgroup_id_1"
  member_id="&MEMBER_ID"
  identifier="Accessories"
  markfordelete="0"
/>
```

catgroup_id è il numero di riferimento interno del gruppo di catalogo. In WebSphere Commerce, a ciascun gruppo di catalogo viene assegnato un numero di riferimento interno che identifica il gruppo quando vengono aggiunte voci di catalogo. identifier è un nome interno per il gruppo di catalogo. Entrambi gli elementi sono univoci all'interno delle risorse database e non possono essere duplicati.

I nomi e le descrizioni appartengono al file catalog.xml specifico della locale, di cui ne è richiesto uno per ciascuna locale supportata dal negozio. Un tipico gruppo di catalogo contenente le informazioni traducibili appare come il seguente estratto:

```
<catgrpdesc
  language_id="&en_US"
  catgroup_id="@catgroup_id_1"
  name="Accessories"
  shortdescription="Accessories"
  longdescription="Accessories"
  published="1"
/>
```

language_id identifica la lingua in cui sono riportate le informazioni sul catalogo. Questo identificativo deve essere modificato in modo da corrispondere a ciascuna lingua supportata dal negozio. Il valore di name viene visualizzato al cliente insieme agli elementi shortdescription e longdescription che possono contenere una descrizione rispettivamente breve e dettagliata del gruppo di catalogo.

Quando viene creato un nuovo gruppo di catalogo, seguire le istruzioni sopra riportate per le informazioni.

Note:

1. Mentre gli elementi identifier e name nell'esempio sopra riportato sono identici, il contenuto può variare. Ad esempio, è possibile scegliere di rinominare il gruppo di catalogo in **Aggiunte complementari**. In tal caso, non è necessario modificare le informazioni di identifier, ma solo quelle di name.
2. Quando si eliminano gruppi di catalogo, assicurarsi che le ricorrenze di catgroup_id vengano appropriatamente aggiornate. Ad esempio, se si desidera eliminare anche le voci di catalogo del gruppo di catalogo, è necessario rimuovere tutte le voci XML. Tuttavia, se si desidera conservare le voci di catalogo, è necessario trasformare catgroup_id nel gruppo corretto.

Voci di catalogo

Le voci di catalogo vengono create in un catalogo WebSphere Commerce utilizzando le informazioni delle tabelle di database CATENTRY e CATENTDESC. Una voce di catalogo può essere un prodotto, un articolo, un pacco, un bundle o un kit dinamico. Dal file `catalog.xml`, una tipica voce di catalogo è simile al seguente estratto:

```
<catentry
  catentry_id="@product_id_102"
  baseitem_id="@baseitem_id_102"
  member_id="MEMBER_ID"
  catenttype_id="ProductBean"
  partnumber="product-sku-nf-102"
  mfpartmentnumber="product-sku-nf-102"
  mfname="NewFashion"
  markfordelete="0"
  buyable="1"
/>
```

`catentry_id` è il numero di riferimento interno della voce di catalogo del prodotto. `baseitem_id` è l'articolo di base a cui si riferisce la voce per motivi di inventario. `partnumber` è il numero di riferimento che identifica il numero di parte della voce di catalogo. `mfpartmentnumber` è il numero di parte utilizzato dal produttore per identificare la voce di catalogo. Questi elementi sono univoci all'interno delle risorse database e non possono essere duplicati.

`catenttype_id` identifica il tipo di voce di catalogo: `ItemBean`, `ProductBean`, `PackageBean`, `BundleBean` o `DynamicKitBean`.

I nomi e le descrizioni appartengono al file `catalog.xml` specifico della locale, di cui ne è richiesto uno per ciascuna locale supportata dal negozio. In questo file sono incluse anche le immagini che raffigurano la merce. Un tipico gruppo di catalogo contenente le informazioni traducibili appare come il seguente estratto:

```
<catentdesc
  catentry_id="@product_id_102"
  language_id="en_US"
  name="Belt"
  shortdescription="Classic belt"
  longdescription="This classic belt looks great with your favorite jeans,
  or takes you to work in style. 1 1/2 inches wide in full-grain leather
  with a solid nickel buckle."
  thumbnail="images/mens_accessories_belt_sm.gif"
  fullimage="images/mens_accessories_belt.gif"
  available="1"
  published="1"
/>
```

`language_id` identifica la lingua in cui sono riportate le informazioni sul catalogo. Questo identificativo deve essere modificato in modo da corrispondere a ciascuna lingua supportata dal negozio. Il valore di `name` viene visualizzato al cliente insieme agli elementi `shortdescription` e `longdescription` che possono contenere una descrizione rispettivamente breve e dettagliata della voce catalogo.

Quando viene creata una voce di catalogo, seguire le istruzioni sopra riportate per le informazioni.

Note:

1. Quando si eliminano voci di catalogo, assicurarsi che ciascuna ricorrenza degli elementi univoci venga appropriatamente aggiornata. Ad esempio, se si desidera eliminare le voci di catalogo del gruppo di catalogo, è necessario

rimuovere tutte le voci XML. Tuttavia, se si desidera conservare le voci di catalogo, è necessario trasformare `catgroup_id` nel gruppo corretto.

2. I prodotti devono essere creati prima di altri tipi di voci di catalogo.

Se non si desidera modificare manualmente i file XML, utilizzare gli strumenti di gestione prodotti o le utilità di Catalog Manager.

Strumenti di gestione prodotti

Gli strumenti di gestione prodotti in WebSphere Commerce Accelerator consentono di gestire i prodotti del catalogo principale del negozio con diverse procedure guidate e blocchi appunti. E' possibile aggiornare il contenuto del catalogo o creare nuovi dati di catalogo:

- Creare, aggiornare ed eliminare prodotti e dettagli sui prodotti con la procedura guidata o il blocco appunti. I prodotti funzionano come modelli per le SKU, i singoli articoli che sono infine venduti a un cliente. I dettagli sui prodotti includono il codice del prodotto (che identifica in maniera univoca il prodotto), il nome e la descrizione del prodotto, le opzioni di commercializzazione (ad esempio visualizzare un prodotto ai clienti o indicare che il prodotto fa parte di una speciale promozione), le immagini dei prodotti, le indicazioni di spedizione e tassazione, gli sconti assegnati ai prodotti e le informazioni sul produttore.
- Generare, aggiornare ed eliminare le SKU (o articoli) per gli acquisiti. Le SKU rappresentano ogni articolo ordinabile della merce in vendita. Tutte le SKU associate ad un particolare prodotto presentano la stessa serie di attributi e si differenziano per i valori di questi attributi. Aggiunte o modifiche apportate alle SKU includono le stesse informazioni dei prodotti, eccetto quelle relative agli ordini.
- Creare, aggiornare o eliminare categorie (o gruppi di catalogo), che sono gruppi di oggetti con proprietà simili utilizzati per organizzare i prodotti o i servizi offerti dal negozio. E' possibile gestire la gerarchia di categorie del catalogo principale creando, modificando ed eliminando le categorie e i dettagli sulle categorie, ad esempio il codice della categoria, il nome e la descrizione, incluso la categoria parent e le immagini.
- Associare i prodotti e le SKU alle categorie scegliendo la categoria parent o spostando i prodotti e le SKU da una categoria all'altra.
- Creare attributi e valori di attributi per i prodotti. Ogni possibile combinazione di attributi e valori di attributi equivale ad una nuova SKU. E' necessario definire i valori dell'attributo prima di assegnarli alle SKU. Dopo aver creato gli attributi e i relativi valori, è possibile creare o aggiornare le informazioni quali il nome, la descrizione (testo, numeri interi o numeri decimali) e la sequenza in cui appaiono attributi e valori.
- Creare, aggiornare, eliminare e associare l'assegnazione prezzi del catalogo ai prodotti. E' possibile definire un prezzo di un prodotto o SKU, in una o più valute, insieme a una serie di condizioni, ad esempio impostare un prezzo per quantità singole o collettive, che devono essere soddisfatte per applicare un determinato prezzo.

Per istruzioni dettagliate su ciascuna attività è possibile consultare la sezione relativa alla gestione dei prodotti della guida in linea.

Note:

1. E' consigliabile utilizzare gli strumenti di gestione solo per modifiche di lieve entità. Per aggiornamenti di una certa importanza, ad esempio l'aggiunta o la rimozione della merce stagionale o la preparazione di una svendita, utilizzare il pacchetto Loader.

2. Qualsiasi modifica apportata ai dati del catalogo non può essere visualizzata nel negozio se non viene disabilitata la memorizzazione cache o non vengono rimosse le pagine JSP correnti. Per ulteriori informazioni, fare riferimento al comando CacheDelete nella guida in linea di WebSphere Commerce. Il comando CacheDelete avvia la rimozione remota della cache delle pagine dinamiche e consente di gestire la cache senza richiedere l'accesso diretto al file system. Prima di utilizzare questo comando, assicurarsi che sia abilitata l'opzione Invalidazione automatica della pagina. Per utilizzare questo comando, è necessario collegarsi come responsabile del sito o del negozio.

Catalog Manager

E' anche possibile gestire il catalogo mediante il pacchetto Loader o Web Editor, due programmi di utilità disponibili da Catalog Manager. Il pacchetto Loader è l'ideale per l'importazione di grandi quantità di informazioni esistenti sul prodotto nel database. In WebSphere Commerce, questo è lo strumento principale per creare e gestire le informazioni del catalogo. Questo pacchetto è composto principalmente da utilità comandi per la preparazione e il caricamento dei dati in un database WebSphere Commerce. Il pacchetto Loader consente anche di estrarre i dati da un database come documento XML, convertire i dati XML in formati XML alternativi e convertire i dati da formato variabile delimitato da caratteri in formato XML.

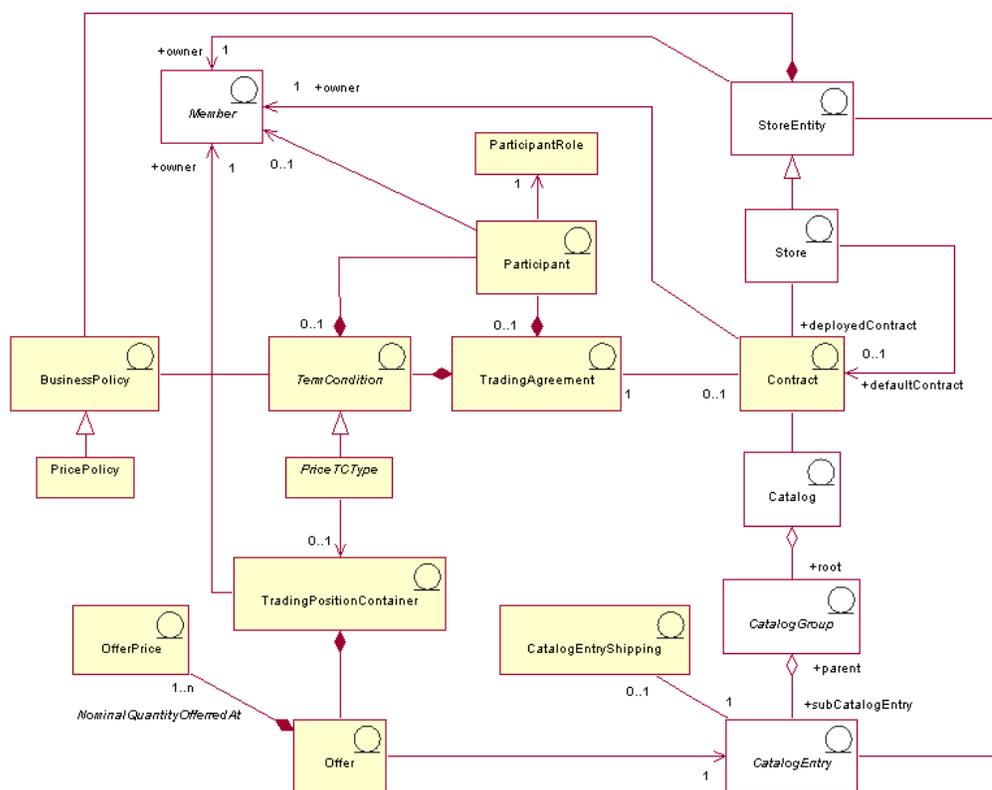
Web Editor utilizza un'interfaccia browser Web in cui è possibile creare, modificare o eliminare i dati di catalogo. I formati dei dati per la visualizzazione e l'aggiornamento delle informazioni sono fondamentali in Web Editor. Nel più semplice dei casi, i formati corrispondono alle tabelle del database WebSphere Commerce. Il responsabile può scegliere di utilizzare i formati predefiniti o di personalizzare i formati disponibili. Per ulteriori informazioni, fare riferimento a *IBM WebSphere Commerce Version 5.4 Catalog Manager - Guida per l'utente*.

Capitolo 9. Risorse assegnazione prezzi

L'assegnazione prezzi rappresenta il prezzo di una voce di catalogo e i criteri che devono essere soddisfatti per l'utilizzo di quel prezzo. Per creare un catalogo funzionale, è necessario aggiungere informazioni sull'assegnazione prezzi al database. E' possibile creare le informazioni sui prezzi in formato file XML che possono essere caricati nel database mediante il pacchetto Loader. In alternativa, è possibile utilizzare gli strumenti di gestione dei prodotti di WebSphere Commerce Accelerator per piccole quantità di dati sui prezzi.

Assegnazione prezzi in WebSphere Commerce

Il grafico di seguito riportato illustra le risorse assegnazione prezzi in WebSphere Commerce Server.



Questo grafico, insieme a tutti gli altri grafici della sezione relativa ai dati del negozio fa parte del modello di informazioni di WebSphere Commerce Server. Per ulteriori informazioni sul modello di informazioni, fare riferimento alla sezione "Modello di informazioni dati del negozio" a pagina 27. Per ulteriori informazioni sulle convenzioni utilizzate in questo grafico, fare riferimento all'Appendice A, "Legenda UML" a pagina 317.

Offerta

Le *offerte*, o assegnazione prezzi, sono prezzi diversi per lo stesso prodotto o articolo stabiliti per clienti o organizzazioni diverse. Un'offerta rappresenta il prezzo di una voce di catalogo ed i criteri, quali la quantità da acquistare, che il

cliente deve soddisfare per pagare quel prezzo. Ad esempio, merci e servizi hanno spesso prezzi diversi per bambini, studenti, adulti e anziani. In WebSphere Commerce, un'offerta è nota anche come posizione commerciale ed è parte di un contenitore di posizioni commerciali.

Prezzo offerta

Il *prezzo offerta* è il prezzo al quale le voci di catalogo vengono offerte da un negozio mediante accordi commerciali o contratti. Per un'offerta possono essere definiti uno o più prezzi in diverse valute.

Contenitore di posizioni commerciali

Un'offerta è parte di un *contenitore di posizioni commerciali*, di proprietà di un membro. Un contenitore di posizioni commerciali contiene varie posizioni commerciali. Può essere disponibile per tutti i clienti oppure per quelli appartenenti a gruppi selezionati mediante gli accordi commerciali o i contratti e i termini e le condizioni dei contratti. In un contratto, un contenitore di posizioni commerciali è un business object prezzi a cui possono fare riferimento diverse politiche aziendali sui prezzi che possono essere condivise da un negozio o da tutti i negozi di un gruppo di negozi. Un contenitore di posizioni commerciali è anche noto come *listino prezzi*.

Termini e condizioni

I *termini e le condizioni* definiscono il comportamento e le proprietà di un accordo commerciale. Poiché molti aspetti del funzionamento di un negozio vengono definiti da politiche aziendali, molti termini e condizioni fanno riferimento alle politiche aziendali.

Termini e condizioni per l'attribuzione prezzi

I *termini e condizioni per l'attribuzione prezzi* definiscono i prodotti disponibili sotto un contratto e i prezzi che il cliente pagherà per i prodotti. In un contratto è richiesto almeno uno dei seguenti termini sui prezzi:

- Un *listino prezzi personalizzato* specifica che, in un contratto, sono stati personalizzati sia l'elenco dei prodotti in vendita che i prezzi relativi. Gli articoli non sono limitati ad una sezione del catalogo negozio ma possono provenire da qualsiasi parte del catalogo.
- Un *intero catalogo con termini di rettifica* offre tutti i prodotti disponibili per la vendita in un catalogo negozio con una rettifica percentuale (ricarico o sconto) a partire dal prezzo base, come definito nel catalogo negozio. Se non viene specificata alcuna rettifica, gli articoli sono venduti al prezzo base.
- Un *listino prezzi con termini di rettifica* offre tutti i prodotti disponibili per la vendita in un listino prezzi con una rettifica percentuale (ricarico o sconto) a partire dal prezzo base, come definito nel catalogo negozio. Se non viene specificata alcuna rettifica, gli articoli sono venduti al prezzo base.
- Un *listino prezzi personalizzato* è simile al listino prezzi con rettifiche ad eccezione del fatto che la rettifica non è applicata a tutto il listino prezzi. La rettifica viene apportata ad un sottoinsieme del listino prezzi. Il sottoinsieme del listino prezzi può essere o una politica aziendale di un gruppo di prodotti o un gruppo di prodotti personalizzato.

Accordo commerciale

Un *accordo commerciale* può essere un contratto, un RFQ, un conto commerciale o un'asta. Un accordo commerciale è un accordo negoziato tra un rivenditore e un

compratore per cui il compratore è abilitato ad acquistare alcuni articoli con i termini e le condizioni specificate e le politiche aziendali stipulate nel contratto. Ad esempio, un contratto consente al cliente di acquistare i prodotti di un negozio al prezzo specificato, per un determinato periodo di tempo, in base ai termini e alle condizioni sull'assegnazione dei prezzi. In WebSphere Commerce, tutti i clienti devono fare acquisti in un negozio sotto contratto, un negozio può distribuire uno o più contratti e uno di questi contratti può essere scelto come contratto predefinito. Un contratto predefinito contiene un insieme di termini e condizioni associati ad un insieme di politiche predefinite del negozio. Un accordo commerciale può contenere zero o più partecipanti con diversi ruoli.

Partecipante

Un *partecipante* può essere parte di un accordo commerciale o dei termini e condizioni. Un partecipante è un membro di un gruppo di membri, di un'organizzazione e così via. Se per un contratto è specificato un partecipante con il ruolo di acquirente, l'acquirente deve essere membro del partecipante acquirente per fare acquisti sotto contratto. I termini e condizioni del contratto possono inoltre contenere zero o più partecipanti.

Ruolo partecipante

Un partecipante può avere uno dei seguenti *ruoli partecipante*:

- Creatore
- Rivenditore
- Acquirente
- Fornitore
- Responsabile approvazioni
- Proprietario account
- Contatto acquirente
- Contatto venditore
- Avvocato
- Amministratore

Contratto

Un *contratto* contiene l'offerta per il prodotto. In WebSphere Commerce, tutti i clienti devono fare acquisti sotto contratto. Un contratto consente al cliente di acquistare i prodotti di un negozio al prezzo specificato, per un determinato periodo di tempo, in base ai termini, alle condizioni e alle politiche aziendali stipulate nel contratto. Un negozio presenta zero o più contratti ed almeno un contratto predefinito.

Politica aziendale

Le *politiche aziendali* sono insiemi di regole seguite da un negozio o da un gruppo di negozi che definiscono processi commerciali, pratiche industriali oltre all'ambito e alle caratteristiche delle offerte di un negozio o di un gruppo di negozi. Le politiche aziendali sono rafforzate con una combinazione di uno o più comandi della politica aziendale che implementano le regole di tale politica, di un riferimento all'oggetto commerciale su cui operano le regole e di un insieme di proprietà per la configurazione del funzionamento dei comandi della politica aziendale.

Politica prezzi

Una *politica prezzi* contiene un riferimento ad un listino prezzi e può essere associata a diversi comandi relativi alla politica aziendale che definiscono il modo in cui le politiche aziendali verranno utilizzate per i listini prezzi. La politica può essere definita per un negozio o un gruppo di negozi. Se la politica è registrata per un gruppo di negozi, può essere utilizzata da tutti i negozi di quel gruppo.

Spedizione delle voci di catalogo

Le informazioni sulla *spedizione delle voci di catalogo* includono le informazioni sul modo in cui il prodotto viene confezionato per la spedizione. Per ciascuna voce di catalogo possono essere definiti diversi tipi di informazioni sulla spedizione. Ad esempio, l'altezza, il peso e la lunghezza del prodotto confezionato.

Altre risorse assegnazione prezzi

Le risorse di seguito riportate sono associate all'assegnazione prezzi:

- Un *membro* proprietario del contenitore di posizioni commerciali. Un contenitore di posizioni commerciali ha un unico proprietario.
- Un' *entità negozio* che rappresenta un negozio in un database di WebSphere Commerce Server.
- Un *catalogo* contenente le voci di catalogo a cui si fa riferimento in un contratto. Il catalogo contiene tutte le informazioni gerarchiche e per l'esplorazione relative al catalogo in linea e rappresenta un insieme di gruppi di catalogo o articoli di catalogo disponibile per la visualizzazione e l'acquisto presso un negozio in linea.
- Un *gruppo di catalogo* o categoria, vale a dire un raggruppamento generico di voci di catalogo, creato per scopi esplorativi e di divisione dei cataloghi. Un gruppo di catalogo appartiene ad un catalogo e può contenere più di un gruppo di catalogo o di voci di catalogo. E' possibile associare gruppi di catalogo a più di un catalogo.
- Una *voce di catalogo* che rappresenta la merce ordinabile presente in un catalogo in linea. Le voci di catalogo appartengono ai gruppi di catalogo. Un'offerta è sempre associata ad una voce di catalogo.



Per informazioni più dettagliate sulla struttura delle risorse assegnazione prezzi in WebSphere Commerce Server, fare riferimento ai modelli di oggetti e dati di assegnazione prezzi nella guida in linea di WebSphere Commerce.

Creazione di risorse assegnazione prezzi in WebSphere Commerce

Esistono due modi per creare le risorse assegnazione prezzi:

- Creare i prezzi utilizzando gli strumenti per la gestione dei prodotti di WebSphere Commerce Accelerator. Gli strumenti di WebSphere Commerce Accelerator sono adatti soprattutto alla creazione dei prezzi di un catalogo molto piccolo.
- Creare i prezzi in un file XML, che può essere caricato dal Pacchetto WebSphere Commerce Loader, oppure come parte di un archivio negozio, che può essere pubblicato mediante Strumenti del negozio. Questo metodo è particolarmente adatto alla creazione di grandi quantità di dati.

Per ulteriori informazioni sulla creazione dei prezzi mediante gli strumenti per la gestione dei prodotti di WebSphere Commerce Accelerator, fare riferimento alla

guida in linea di WebSphere Commerce. Per ulteriori informazioni sulla creazione dei prezzi in un file XML, fare riferimento alla sezione “Creazione di risorse assegnazione prezzi in un file XML”.

Creazione di risorse assegnazione prezzi in un file XML

Creare le risorse assegnazione prezzi in formato file XML che possono essere caricati nel database mediante il pacchetto Loader. Per ulteriori informazioni sul pacchetto Loader, consultare la Parte 7, “Pubblicazione del negozio” a pagina 213.

1. Rivedere i file XML utilizzati per creare le risorse assegnazione prezzi nei negozi di esempio. Tutti i file per i negozi di esempio si trovano nel file dell’archivio negozio corrispondente.

I file dell’archivio negozio si trovano nelle seguenti directory:

-  NT unità:\WebSphere\CommerceServer\samplestores
-  2000 unità:\Program Files\WebSphere\CommerceServer\samplestores
-  AIX /usr/WebSphere/CommerceServer/samplestores
-  Solaris /opt/WebSphere/CommerceServer/samplestores
-  Linux /opt/WebSphere/CommerceServer/samplestores
-  400 /qibm/proddata/WebCommerce/samplestores

Nota: La guida in linea di WebSphere Commerce contiene le informazioni su ciascuna delle risorse dati contenute nei negozi di esempio.

Ciascun negozio di esempio presenta due file `offering.xml`, contenenti le informazioni sull’assegnazione prezzi. Per visualizzare i file `offering.xml` nell’archivio negozio, decomprimere il file con un programma ZIP. I file `offering.xml` si trovano nella directory dei dati. Il file `offering.xml` specifico della lingua si trova in una sottodirectory specifica delle locale della directory dei dati.

2. Fare riferimento alle informazioni nell’Appendice B, “Creazione di dati” a pagina 319.
3. Creare un file `offering.xml`, copiando uno dei file `offering.xml` presenti negli archivi negozio di esempio o creandone uno nuovo. Per ulteriori informazioni, fare riferimento al file DTD corrispondente a `offering.xml`. I file DTD si trovano nelle seguenti directory:

-  NT unità:\WebSphere\CommerceServer\xml\sar
-  2000 unità:\Program Files\WebSphere\CommerceServer\xml\sar
-  AIX /usr/WebSphere/CommerceServer/xml/sar
-  Solaris /opt/WebSphere/CommerceServer/xml/sar
-  Linux /opt/WebSphere/CommerceServer/xml/sar
-  400 /qibm/proddata/WebCommerce/xml/sar

4. Creare un contenitore di posizioni commerciali. Per offrire dei prezzi per le merci del negozio, è necessario creare prima un contenitore di posizioni commerciali. Per creare un contenitore di posizioni commerciali, aggiungere informazioni alla tabella TRADEPOSCN.
 - a. Utilizzando il seguente esempio come guida, creare un contenitore di posizioni commerciali nel file XML per la tabella TRADEPOSCN:

```
<tradeposcn
tradeposcn_id="@tradeposcn_id_101"
member_id="@MEMBER_ID"
markfordelete="0"
name="ToolTech"
precedence="0"
```

/>

dove

- tradeposcn_id è una chiave univoca generata.
 - member_id è il proprietario del contenitore di posizioni commerciali.
 - markfordelete presenta i seguenti valori:
 - 0 = il TradingPositionContainer può essere utilizzato
 - 1 = il TradingPositionContainer è stato contrassegnato per la cancellazione (consultare il programma di utilità DBClean) e non può essere utilizzato
 - name è un nome mnemonico per il contenitore di posizioni commerciali, univoco per un determinato proprietario.
 - precedence è il valore di precedenza relativo ai contenitori di posizioni commerciali. Quando più contenitori di posizioni commerciali sono qualificati in un determinato momento, viene utilizzato quello con il valore PRECEDENCE maggiore.
5. Associare il catalogo principale al contenitore di posizioni commerciali aggiungendo informazioni alla tabella CATGRPTPC. Quando si associa il catalogo principale al contenitore di posizioni commerciali, ciascuna voce di catalogo del catalogo principale deve avere un prezzo standard. Per ulteriori informazioni sulla creazione dei cataloghi principali, fare riferimento alla sezione “Visualizzazione delle risorse catalogo del negozio” a pagina 77.
- a. Utilizzando il seguente esempio come guida, associare il catalogo principale al contenitore di posizioni commerciali aggiungendo informazioni alla tabella CATGRPTPC:

```
<catgrptpc
catalog_id="@catalog_id_1"
tradeposcn_id="@tradeposcn_id_101"
/>
```

dove

- catalog_id è catalogo principale.
 - tradeposcn_id è il contenitore di posizioni commerciali.
6. Creare offerte e relativi prezzi per le voci di catalogo aggiungendo informazioni alle tabelle OFFER e OFFERPRICE
- a. Utilizzando il seguente esempio come guida, creare un’offerta per una voce di catalogo aggiungendo informazioni alla tabella OFFER. Prima di creare i prezzi, è necessario creare le voci di catalogo. Per ulteriori informazioni sulla creazione di voci di catalogo, fare riferimento alla sezione “Visualizzazione delle risorse catalogo del negozio” a pagina 77.

```
offer
offer_id="@offer_id_138"
startdate="2000-06-19 00:00:00.000000"
catentry_id="@product_id_102"
precedence="0"
published="1"
```

```
identifier="1"  
flags="1"  
tradeposcn_id="@tradeposcn_id_101"  
</>
```

dove

- offer_id è una chiave univoca generata.
 - startdate è l'inizio dell'intervallo di tempo in cui l'offerta è valida.
 - catentry_id è la voce di catalogo offerta per la vendita.
 - precedence è il valore di precedenza relativo alle offerte. Quando più offerte sono qualificate in un determinato momento, viene utilizzata quella con il valore PRECEDENCE maggiore.
 - published è
 - 0 = non pubblicata (temporaneamente disabilitata)
 - 1 = pubblicata
 - 2 = contrassegnata per la cancellazione (e non pubblicata)
 - identifier è un numero che identifica in modo univoco questa offerta insieme alla voce di catalogo e al contenitore di posizioni commerciali specificati.
 - flags sono
 - 1 = shiptoAddressRequired - se 1, il comando OrderPrepare restituisce un errore se un articolo si riferisce a questa offerta ma non dispone di un indirizzo di spedizione.
 - tradeposcn_id è il contenitore di posizioni commerciali di cui fa parte questa offerta.
- b. Utilizzando il seguente esempio come guida, creare un prezzo offerta per una voce di catalogo aggiungendo informazioni alla tabella OFFERPRICE. Il prezzo offerta è il prezzo effettivo al quale una voce di catalogo è offerta per la vendita. Prima di creare i prezzi, è necessario creare le voci di catalogo. Per ulteriori informazioni sulla creazione di voci di catalogo, fare riferimento alla sezione "Visualizzazione delle risorse catalogo del negozio" a pagina 77.

```
<offerprice  
offer_id="@offer_id_138"  
currency="USD"  
price="590.00"  
>
```

dove

- offer_id è l'offerta associata a quel prezzo.
- currency è la valuta in cui è offerto il prezzo.
- price è il prezzo per la quantità nominale (consultare CATENTSHIP.NOMINALQUANTITY) del prodotto a cui si riferisce l'offerta.

Nota: Per visualizzare più valute nel negozio, creare una voce XML diversa nella tabella OFFERPRICE per ciascun valuta. Ad esempio, per visualizzare la valuta in dollari canadesi, utilizzare currency="CAD" in una nuova voce XML. Il valore di price varia per riflettere il prezzo in dollari canadesi. In alternativa, è possibile utilizzare una conversione, il che consente al cliente di visualizzare i tassi diversi

per la valuta selezionata. Per ulteriori informazioni, consultare la sezione "Creazione di risorse valuta mediante un file XML" a pagina 125.

c. Ripetere i passi a e b per tutte le voci del catalogo.



Per ulteriori informazioni sull'uso di @ e &, fare riferimento all'Appendice B, "Creazione di dati" a pagina 319.

Capitolo 10. Risorse contratti

In WebSphere Commerce, tutti i clienti del negozio devono fare shopping sotto contratto. Un contratto consente ai clienti di acquistare i prodotti di un negozio ad un prezzo specificato, per un determinato periodo di tempo, in base a determinate condizioni. Sfogliando il catalogo di un negozio, i clienti possono visualizzare solo i prodotti coperti dai contratti che hanno con il negozio.

Per consentire ai clienti che non hanno alcun contratto con il negozio (ad esempio gli acquirenti ospiti) di fare shopping nel negozio o a tutti i clienti di acquistare anche prodotti non coperti dai contratti in loro possesso, è necessario che il negozio disponga di un *contratto predefinito*.

Importante

WebSphere Commerce Professional Edition supporta solo il contratto predefinito del negozio.

I contratti diversi da quello predefinito del negozio sono supportati solo da WebSphere Commerce Business Edition.

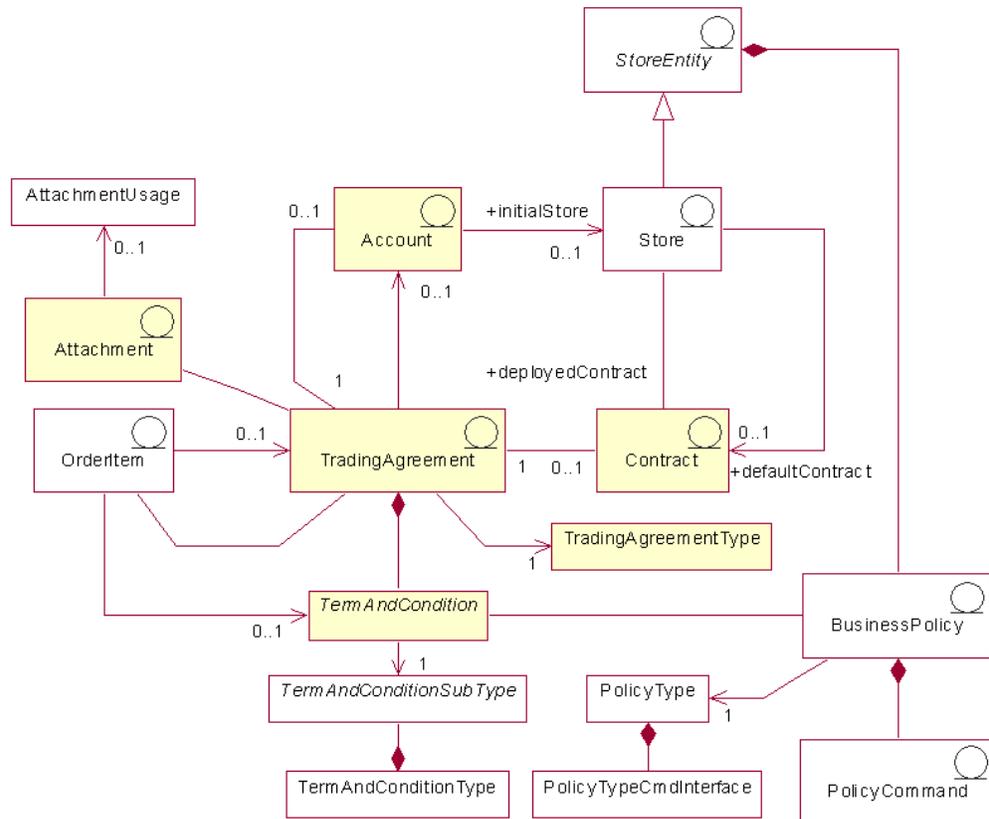
Per consentire a tutti i clienti di fare shopping in un negozio, un negozio creato con WebSphere Commerce deve includere i seguenti elementi:

- Politiche aziendali
- Contratto predefinito

Alle politiche aziendali si fa riferimento nel contratto predefinito, consentendo in tal modo ai clienti di fare shopping in un negozio.

Contratti in WebSphere Commerce

Il grafico di seguito riportato illustra la struttura dei contratti in WebSphere Commerce:



Questo grafico, insieme a tutti gli altri grafici della sezione relativa ai dati del negozio, fa parte del modello di informazioni di WebSphere Commerce Server. Per ulteriori informazioni sul modello di informazioni, fare riferimento alla sezione "Modello di informazioni dati del negozio" a pagina 27. Per ulteriori informazioni sulle convenzioni utilizzate in questo grafico, fare riferimento all'Appendice A, "Legenda UML" a pagina 317.

Conto (conto commerciale)

Un conto commerciale rappresenta la relazione tra un'organizzazione acquirente e un'organizzazione venditrice. Un conto commerciale può essere utilizzato per organizzare altri accordi commerciali e per specificare i termini e le condizioni relativi alla relazione tra un acquirente e un venditore, ad esempio la personalizzazione della fattura, la verifica dell'ordine di acquisto o la gestione della linea di credito di un acquirente con il venditore.

I contratti sono associati ai conti commerciali, in quanto rappresentano un accordo tra un acquirente e un venditore. L'unica eccezione è rappresentata dal contratto predefinito del negozio che non può essere associato ad un conto commerciale. Ad un conto commerciale possono essere associati diversi contratti.

Un conto commerciale rappresenta un tipo di accordo commerciale. Per una descrizione degli accordi commerciali, fare riferimento alla sezione "Accordo commerciale" a pagina 99.

Importante: I conti commerciali sono supportati solo da WebSphere Commerce Business Edition.

Contratto

I contratti attivi associati ai negozi sono di due tipi: i contratti predefiniti e i contratti distribuiti. I contratti distribuiti appartengono a specifiche organizzazioni di acquirenti o ai singoli acquirenti e possono essere creati con WebSphere Commerce Accelerator dopo aver creato il negozio. Un contratto distribuito è associato a un conto commerciale. Un contratto predefinito definisce il comportamento predefinito del negozio rispetto agli acquirenti che non hanno alcun contratto con il negozio stesso. Un contratto predefinito può essere creato solo utilizzando file XML e per un negozio è possibile definire un solo contratto predefinito. Per ulteriori informazioni sui contratti, fare riferimento alle informazioni in linea. Per informazioni sulla creazione della risorsa contratto predefinito, fare riferimento alla sezione “Creazione di una risorsa contratto predefinita in WebSphere Commerce” a pagina 101.

Un contratto rappresenta un tipo di accordo commerciale. Per una descrizione degli accordi commerciali, fare riferimento alla sezione “Accordo commerciale”.

Accordo commerciale

WebSphere Commerce fornisce diversi meccanismi commerciali che governano le interazioni tra acquirenti e venditori. I seguenti meccanismi commerciali sono supportati da diverse edizioni di WebSphere Commerce:

- Aste (supportate dalle edizioni Business e Professional)
- Conti commerciali (supportati solo dalla edizione Business)
- Contratti (vedere le limitazioni discusse precedentemente in questo capitolo)
- RFQ (Request For Quotes) (supportate solo dalla edizione Business)

Tutti questi meccanismi commerciali hanno proprietà comuni. Ad esempio, tutti i meccanismi commerciali hanno partecipanti e regole. Le regole che governano il comportamento dei meccanismi commerciali sono note come *termini e condizioni* in WebSphere Commerce.

Un accordo commerciale rappresenta un'istanza di un meccanismo commerciale e registra le proprietà di quell'istanza. Ciascun contratto, conto commerciale e RFQ in WebSphere Commerce è rappresentato da un accordo commerciale. Un unico accordo commerciale presiede a tutte le aste in WebSphere Commerce.

Un accordo commerciale è costituito da un profilo presente nella tabella TRADING, da partecipanti presenti nella tabella PARTICIPANT, da termini e condizioni presenti nella tabella TERMCOND e da allegati facoltativi presenti come URI (Universal Resource Identifiers) nella tabella ATTACHMENT. Poiché un accordo commerciale può avere più allegati, questi ultimi sono connessi al relativo accordo commerciale mediante la tabella TRDATTACH. Gli allegati non sono supportati dalle richieste RFQ.

Oltre all'accordo commerciale generale, ciascun tipo di accordo commerciale presenta informazioni supplementari specifiche nella relativa tabella: CONTRACT contiene informazioni specifiche sui contratti, RFQ contiene informazioni specifiche sulle risposte RFQ e ACCOUNT contiene informazioni specifiche sul conto commerciale.

Termini e condizioni

I termini e le condizioni definiscono il comportamento e le proprietà di un accordo commerciale.

Per i contratti, i termini e le condizioni definiscono come viene implementato un contratto per un'organizzazione acquirenti. Definiscono cosa viene venduto in base al contratto, il prezzo degli articoli in vendita, le modalità di spedizione e di pagamento degli ordini, il modo in cui vengono gestite le restituzioni degli articoli, il modo in cui gli ordini vengono approvati e il luogo da cui vengono spediti.

Poiché molti aspetti del funzionamento di un negozio vengono definiti da politiche aziendali, molti termini e condizioni fanno riferimento alle politiche aziendali. I termini e le condizioni forniscono i parametri per le politiche aziendali a cui fanno riferimento. Grazie ai parametri è possibile modificare il comportamento delle politiche aziendali per ciascun contratto.

Politiche aziendali

Le politiche aziendali costituiscono un insieme di regole stabilite per un negozio o per un gruppo di negozi. Esse definiscono i processi aziendali, le pratiche industriali e l'ambito e le caratteristiche delle offerte di un negozio. Rappresentano l'origine centrale e il modello di riferimento di tutte le pratiche consentite e supportate all'interno di un negozio o gruppo di negozi.

In WebSphere Commerce, le politiche aziendali vengono applicate con una combinazione di uno o più comandi della politica aziendale che implementano le regole di tale politica, un riferimento all'oggetto commerciale su cui operano le regole e un insieme di proprietà per la configurazione del funzionamento dei comandi della politica aziendale. I termini e le condizioni possono fornire i parametri per le politiche aziendali a cui fanno riferimento. Ciò consente di modificare il comportamento della politica aziendale a seconda del termine e condizione di riferimento.

Allegato

Un allegato fornisce i dettagli di un accordo commerciale non specificati da altri elementi dell'accordo. Un esempio di allegato è un file contenente informazioni sui requisiti della RFQ e note generali sulla RFQ. Un accordo commerciale può avere più allegati. Gli allegati vengono memorizzati al di fuori di WebSphere Commerce e l'accordo commerciale memorizza gli URI (Universal Resource Identifiers) negli allegati. Vengono di seguito elencati alcuni esempi di URI:

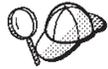
- <http://www.mycompany.com/information/document1.txt>
- <file:///home/joeuser/mydocs/document1>
- <ftp://ftp.mycompany.com/information/attachment.txt>

A tutti gli allegati è possibile assegnare un tipo di uso che indica lo scopo dell'allegato. L'uso dell'allegato è una proprietà opzionale.

Articolo dell'ordine

Un articolo dell'ordine è un prodotto incluso in un ordine. In un singolo ordine è possibile specificare accordi commerciali diversi per articoli diversi. Gli acquirenti possono selezionare l'accordo commerciale in base quale fanno shopping, sia all'inizio del flusso di acquisti che quando aggiungono un articolo all'ordine, a seconda del progetto del negozio. Quando si acquistano articoli sotto i diversi accordi commerciali del contratto, è necessario applicare le seguenti regole:

- Gli accordi commerciali del contratto per tutti gli articoli di un ordine devono condividere almeno un metodo di pagamento. Se ciò non avviene, l'acquirente non può aggiungere l'articolo all'ordine. Solo i metodi di pagamento condivisi da tutti gli articoli di un ordine possono essere utilizzati per pagare l'ordine.
- E' necessario che tutti gli articoli di un ordine provengano da accordi commerciali del contratto appartenenti allo stesso conto commerciale o al contratto predefinito del negozio.



Per informazioni più dettagliate sulla struttura delle risorse contratto in WebSphere Commerce, fare riferimento al modello di dati del contratto nella guida in linea di WebSphere Commerce.

Creazione di una risorsa contratto predefinita in WebSphere Commerce

Il contratto predefinito definisce il comportamento di un negozio. Come per i contratti, è possibile impostare i prodotti disponibili, i prezzi, i metodi di pagamento e di spedizione ed altri comportamenti del negozio.

I contratti predefiniti del negozio forniti con i negozi di esempio di WebSphere Commerce contengono i termini e le condizioni che specificano i seguenti elementi:

- I clienti possono acquistare tutti i prodotti disponibili nel catalogo principale del negozio ai prezzi standard impostati nel catalogo principale (senza sconti o rialzi).
- Le tasse di spedizione sono pagate dal venditore (negozio).
- I clienti possono restituire la merce acquistata entro un determinato numero di giorni senza penali.
- I clienti possono ricevere rimborsi con lo stesso metodo di pagamento utilizzato per effettuare l'acquisto originario.

La versione più generica di contratto predefinito di un negozio omette termini e condizioni che restringono i metodi di pagamento e spedizione utilizzabili dal cliente. In tal modo si consente al cliente di pagare la merce acquistata con un qualsiasi metodo di pagamento predefinito supportato dal negozio e di utilizzare qualsiasi metodo di spedizione disponibile nel negozio.

Le proprietà del contratto predefinito sono definite nei termini e condizioni. Alcuni dei termini e condizioni fanno riferimento alle politiche aziendali. Per ulteriori informazioni sulle politiche aziendali e sui termini e condizioni, fare riferimento alle informazioni in linea.

Per creare una risorsa contratto predefinita, procedere come segue:

1. Esaminare le informazioni in linea relative a termini, condizioni, contratti, contratti predefiniti e politiche aziendali.
2. Esaminare le politiche aziendali definite nel file `wcs.bootstrap.xml`. Per informazioni sul file `wcs.bootstrap.xml`, fare riferimento alle informazioni in linea.
3. Esaminare i file utilizzati per creare le risorse contratto predefinite per i negozi di esempio. Tutti i file dei negozi di esempio risiedono nel file dell'archivio negozio corrispondente. Ogni negozio di esempio include un file `businesspolicy.xml` e un file `contract.xml`, contenenti informazioni dettagliate sulla politica aziendale e sul contratto predefinito. I file degli archivi negozio si trovano nelle seguenti directory:

-  /usr/WebSphere/CommerceServer/samplestores
-  /qibm/proddata/WebCommerce/samplestores
-  /opt/WebSphere/CommerceServer/samplestores
-  /opt/WebSphere/CommerceServer/samplestores
-  *unità*: \Program Files\WebSphere\CommerceServer\samplestores
-  *unità*: \WebSphere\CommerceServer\samplestores

Note:

- a. La guida in linea di WebSphere Commerce contiene le informazioni su ciascuna delle risorse dati contenute nei negozi di esempio.
 - b. Per visualizzare i file `businesspolicy.xml` e `contract.xml` nell'archivio negozio, decomprimerli con un programma ZIP. I file risiedono nella `directory data`.
 - c. I file delle risorse contratto per il negozio di esempio ToolTech fornito con WebSphere Commerce Business Edition includono informazioni relative ai contratti diversi da quello predefinito.
4. Fare riferimento alle informazioni nell'Appendice B, "Creazione di dati" a pagina 319.
 5. Creare un file `businesspolicy.xml` copiando uno dei file `businesspolicy.xml` presenti negli archivi dei negozi oppure creandone uno nuovo. E' possibile trovare le istruzioni sulla creazione di un nuovo file in "Creazione dei file XML di politica aziendale" a pagina 103. Se si desidera creare politiche aziendali diverse da quelle precedentemente descritte, consultare il file DTD che corrisponde a `businesspolicy.xml`. I file DTD si trovano nelle seguenti directory:
 -  /usr/WebSphere/CommerceServer/xml/sar
 -  /qibm/proddata/WebCommerce/xml/sar
 -  /opt/WebSphere/CommercServer/xml/sar
 -  /opt/WebSphere/CommercServer/xml/sar
 -  *unità*: \Program Files\WebSphere\CommerceServer\xml\sar
 -  *unità*: \WebSphere\CommerceServer\xml\sar
 6. Caricare il file `businesspolicy.xml` mediante il pacchetto Loader. Per ulteriori informazioni sul pacchetto Loader, consultare la Parte 7, "Pubblicazione del negozio" a pagina 213. Se il negozio da creare è multiculturale, è possibile creare file XML diversi per ogni locale supportata dal negozio. Il file specifico delle locale deve contenere tutte le informazioni sulla descrizione, in modo che possa essere facilmente tradotto.
 7. Creare un file `contract.xml` copiando uno dei file `contract.xml` presenti negli archivi negozio di esempio o creando un nuovo file. E' possibile trovare le istruzioni sulla creazione di un nuovo file in "Creazione di un file XML di contratti predefiniti" a pagina 104. Se si desidera creare un nuovo contratto predefinito più complesso, esaminare il file `B2BTrading.dtd` che definisce la struttura di un file `contract.xml`. Il file `B2BTrading.dtd` risiede nella seguente directory:
 -  /usr/WebSphere/CommerceServer/xml/trading
 -  /qibm/proddata/WebCommerce/xml/trading

-  /opt/WebSphere/CommerceServer/xml/trading
 -  /opt/WebSphere/CommerceServer/xml/trading
 -  unità:\Program Files\WebSphere\CommerceServer\xml\trading
 -  unità:\WebSphere\CommerceServer\xml\trading
8. Pubblicare il contratto utilizzando il comando `ContractImportApprovedVersion`. Per ulteriori informazioni, consultare il Capitolo 29, “Pubblicazione dei contratti e dei conti commerciali” a pagina 285. Ulteriori dettagli sul comando `ContractImportApprovedVersion` sono disponibili nelle informazioni in linea.

Gli utenti di WebSphere Commerce Business Edition possono definire i contratti per clienti specifici mediante WebSphere Commerce Accelerator. Per ulteriori informazioni sulla creazione dei contratti per clienti specifici, fare riferimento alle informazioni in linea.

Creazione dei file XML di politica aziendale

Nonostante WebSphere Commerce fornisca un certo numero di politiche aziendali a cui fanno riferimento i termini e le condizioni del contratto predefinito del negozio, altre politiche aziendali devono essere definite dall'utente. L'utente deve definire le politiche aziendali relative alle spese di restituzione, all'approvazione della restituzione e all'assegnazione dei prezzi a cui faranno riferimento i termini del contratto predefinito del negozio. I comandi per queste politiche aziendali sono forniti e possono essere utilizzati senza alcuna modifica. Se si desidera creare politiche aziendali personalizzate, fare riferimento a *IBM WebSphere Commerce - Guida per il programmatore*.

Per creare politiche aziendali per il negozio, è necessario creare la politica aziendale e associare a tale politica uno o più comandi. Per creare una politica aziendale, aggiungere informazioni alla tabella POLICY. Per associare un comando ad una politica aziendale, aggiungere informazioni alla tabella POLICYCMD.

Per creare una politica aziendale e associare dei comandi a tale politica, procedere come segue:

1. Creare una politica aziendale nel file XML delle politiche aziendali aggiungendo informazioni alla tabella POLICY. Utilizzare il seguente esempio come guida:

```
<policy
policy_id="@policy_id_10"
policyname="MasterCatalogPriceList"
policytype_id="Price"
storeent_id="@storeent_id_1"
properties="name=Infashion&member_id=MEMBER_ID"
/>
```

dove

- `policy_id` è l'identificativo numerico univoco per la politica aziendale.
- `nomepolitica` è un nome univoco per la politica aziendale.
- `policytype_id` è il tipo di politica da definire. I `policytype_id` validi sono i seguenti:
 - InvoiceFormat
 - Payment
 - Price
 - ProductSet

- ReturnApproval
 - ReturnCharge
 - ReturnPayment
 - ShippingCharge
 - ShippingPayment
 - storeent_id rappresenta il negozio o gruppo di negozi.
 - properties rappresenta un elenco di coppie nome-valore inviato al comando di politica aziendale.
2. Associare un comando a una politica aziendale nel file XML delle politiche aziendali aggiungendo informazioni alla tabella POLICYCMD. Utilizzare il seguente esempio come guida:

```
<polycmd
policy_id="@policy_id_10"
businesscmdclass=
"com.ibm.com.commerce.price.commands.CalculateContractPricesCmdImpl"
/>
```

dove

- policy_id rappresenta l'identificativo numerico della politica aziendale a cui viene associato il comando.
- businesscmdclass è il nome della classe Java che implementa la politica aziendale.

Le interruzioni di riga nell'attributo businesscmdclass sono state inserite solo per facilitare la lettura.



Per ulteriori informazioni sull'uso dei simboli @ e &, fare riferimento all'Appendice B, "Creazione di dati" a pagina 319.

Creazione di un file XML di contratti predefiniti

Per creare un contratto predefinito, è necessario definire il contratto e il proprietario, la descrizione, i partecipanti e i termini e le condizioni del contratto. Le informazioni sul contratto vengono memorizzate in quattro tabelle: CONTRACT, PARTICIPNT, TRADING e TERMCOND.

Il contratto predefinito viene associato a un negozio mediante la tabella STOREDEF. Per gli utenti WebSphere Commerce Business Edition, i contratti diversi da quello predefinito vengono associati a un negozio mediante la tabella di database STORECNTR.

Per creare un contratto predefinito, procedere come segue:

1. Definire il contratto predefinito nel file XML. Il contratto predefinito viene specificato all'inizio del file XML nel modo di seguito riportato:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE Trading SYSTEM "B2BTrading.dtd">
<Trading>
<Contract state="Active" origin="Manual"
name="&STORE_IDENTIFIER; Default Contract" majorVersionNumber="1"
minorVersionNumber="0" contractUsage="Default">
```

Le interruzioni di riga in Contract sono state inserite solo per agevolare la visualizzazione.

2. Definire il proprietario del contratto. Utilizzare il seguente esempio come guida:

```
<ContractOwner>
  <Member>
    <Organization distinguishName="&MEMBER_IDENTIFIER;" />
  </Member>
</ContractOwner>
```

dove `distinguishName` è il nome dell'utente che possiede il contratto in formato nome distinto LDAP. Ad esempio, `uid=erickoeck,ou=People,dc=ibm,dc=com`.

- Definire la descrizione del contratto nel file XML del contratto. Utilizzare il seguente esempio come guida:

```
<ContractDescription title="This is a store default contract." languageId="-1">
</ContractDescription>
```

dove

- `title` è una descrizione del contratto.
- `languageId` è la lingua in cui è scritta la descrizione. I valori di seguito riportati sono predefiniti per `languageId`:
 - -1 (Inglese – USA)
 - -2 (Francese)
 - -3 (Tedesco)
 - -4 (Italiano)
 - -5 (Spagnolo)
 - -6 (Portoghese brasiliano)
 - -7 (Cinese semplificato)
 - -8 (Cinese tradizionale)
 - -9 (Coreano)
 - -10 (Giapponese)

Ulteriori valori possono essere definiti per `languageId` aggiornando le risorse lingua per il negozio. Per ulteriori informazioni sulle risorse lingua, consultare Capitolo 14, "Risorse lingua" a pagina 121.

- Definire i partecipanti del contratto nel file XML del contratto. Utilizzare il seguente esempio come guida:

```
<Participant role="Buyer">
</Participant>
<Participant role="Seller">
  <Member>
    <Organization distinguishName="&MEMBER_IDENTIFIER;" />
  </Member>
</Participant>
```

dove `distinguishName` è il nome dell'utente venditore per questo contratto in formato nome distinto LDAP. Ad esempio, `uid=erickoeck,ou=People,dc=ibm,dc=com`. In molti casi corrisponde al proprietario del contratto.

Nota: Nessun membro è specificato nel ruolo di partecipante acquirente in quanto il contratto è disponibile per tutti gli utenti con ruolo di acquirente.

- Definire i termini e le condizioni nel file XML del contratto. Gli elementi e gli attributi XML sono diversi per i vari tipi di termini e condizioni. Utilizzare il file `B2BTrading.dtd` per individuare gli elementi e gli attributi XML da utilizzare per ciascun tipo di termine. Quando si definiscono i termini e le condizioni, di solito vengono utilizzati i seguenti attributi:

policyName

Il nome della politica commerciale a cui si riferiscono il termine e la condizione. Il nome memorizzato in POLICY.POLICYNAME.

policyType

Il tipi di politica commerciale a cui si riferiscono il termine e la condizione. I valori validi sono:

- Price
- ProductSet
- InvoiceFormat
- Payment
- ReturnApproval
- ReturnCharge
- ReturnPayment
- ShippingCharge
- ShippingMode

storeIdentity

Il negozio o gruppo di negozi per il termine e la condizione.

distinguishName

Il nome dell'utente che possiede il negozio o gruppo di negozi. Il nome deve essere in formato nome distinto LDAP. Ad esempio, uid=wcsadmin,o=Root Organization.

I termini e le condizioni di esempio di seguito riportati sono preceduti da una descrizione dell'elemento che definiscono:

- Tutti gli acquirenti possono acquistare tutti gli articoli presenti nel catalogo principale del negozio ai prezzi impostati in tale catalogo:

```
<TermCondition>
  <PriceTC>
    <PriceTCMasterCatalogWithOptionalAdjustment>
    </PriceTCMasterCatalogWithOptionalAdjustment>
  </PriceTC>
</TermCondition>
```

- Gli acquirenti pagano le tasse di spedizione al venditore:

```
<TermCondition>
  <ShippingTC>
    <ShippingTCShippingCharge>
      <PolicyReference policyName="StandardShippingChargeBySeller"
        policyType="ShippingCharge" storeIdentity="&STORE_IDENTIFIER;">
        <Member>
          <User distinguishName="&MEMBER_IDENTIFIER;">
          </Member>
        </PolicyReference>
      </ShippingTCShippingCharge>
    </ShippingTC>
  </TermCondition>
```

Le interruzioni di riga in PolicyReference sono state inserite solo per agevolare la visualizzazione.

- Gli acquirenti possono restituire i prodotti senza alcuna spesa di restituzione. I prodotti devono essere restituiti entro il numero di giorni definito nella politica aziendale ApprovalByDays:

```
<TermCondition>
  <ReturnTC>
    <ReturnTCReturnCharge>
```

```

<ReturnChargePolicyReference>
  <PolicyReference policyName="NoCharges"
    policyType="ReturnCharge"
    storeIdentity="&STORE_IDENTIFIER;">
    <Member>
      <Organization distinguishName="&MEMBER_IDENTIFIER;">
    </Member>
  </PolicyReference>
</ReturnChargePolicyReference>
<ReturnApprovalPolicyReference>
  <PolicyReference policyName="ApprovalByDays"
    policyType="ReturnApproval"
    storeIdentity="&STORE_IDENTIFIER;">
    <Member>
      <Organization distinguishName="&MEMBER_IDENTIFIER;">
    </Member>
  </PolicyReference>
</ReturnApprovalPolicyReference>
</ReturnTCReturnCharge>
</ReturnTC>
</TermCondition>

```

Le interruzioni di riga nell'elemento PolicyReference sono state inserite solo per agevolare la visualizzazione.

Nota per gli utenti di WebSphere Commerce Business Edition:

Omettere questi termini e condizioni dal contratto predefinito del negozio indica che, per impostazione predefinita, il negozio non accetta restituzioni. Altri contratti, però, consentono agli acquirenti di effettuare restituzioni definendo termini e condizioni della restituzione.

Nota per gli utenti di WebSphere Commerce Professional Edition:

Omettere questi termini e condizioni dal contratto predefinito del negozio indica che il negozio non accetta restituzioni.

- Le restituzioni vengono pagate con lo stesso metodo di pagamento utilizzato dall'acquirente per pagare l'ordine:

```

<TermCondition>
  <ReturnTC>
    <ReturnTCRefundPaymentMethod>
      <PolicyReference policyName="UseOriginalPayment"
        policyType="ReturnPayment" storeIdentity="&STORE_IDENTIFIER;">
        <Member>
          <User distinguishName="&MEMBER_IDENTIFIER;">
        </Member>
      </PolicyReference>
    </ReturnTCRefundPaymentMethod>
  </ReturnTC>
</TermCondition>

```

Le interruzioni di riga in PolicyReference sono state inserite solo per agevolare la visualizzazione.



Per ulteriori informazioni sull'uso dei simboli @ e &, fare riferimento all'Appendice B, "Creazione di dati" a pagina 319.

Capitolo 11. Risorse evasione ordini

I centri di evasione ordini vengono utilizzati dai negozi sia come magazzini di inventario che come centri di spedizione e di ricezione. Un negozio può avere associato uno o più centri di evasione ordini. Il centro di evasione ordini gestisce l'inventario dei prodotti e la spedizione per un negozio. L'evasione ordini include la raccolta, il confezionamento e la spedizione dei prodotti. La raccolta è la selezione dei prodotti in uno o più rilasci da parte di un centro di evasione ordini, il confezionamento è l'inserimento di questi prodotti in contenitori per la spedizione e la spedizione è l'invio ai clienti.

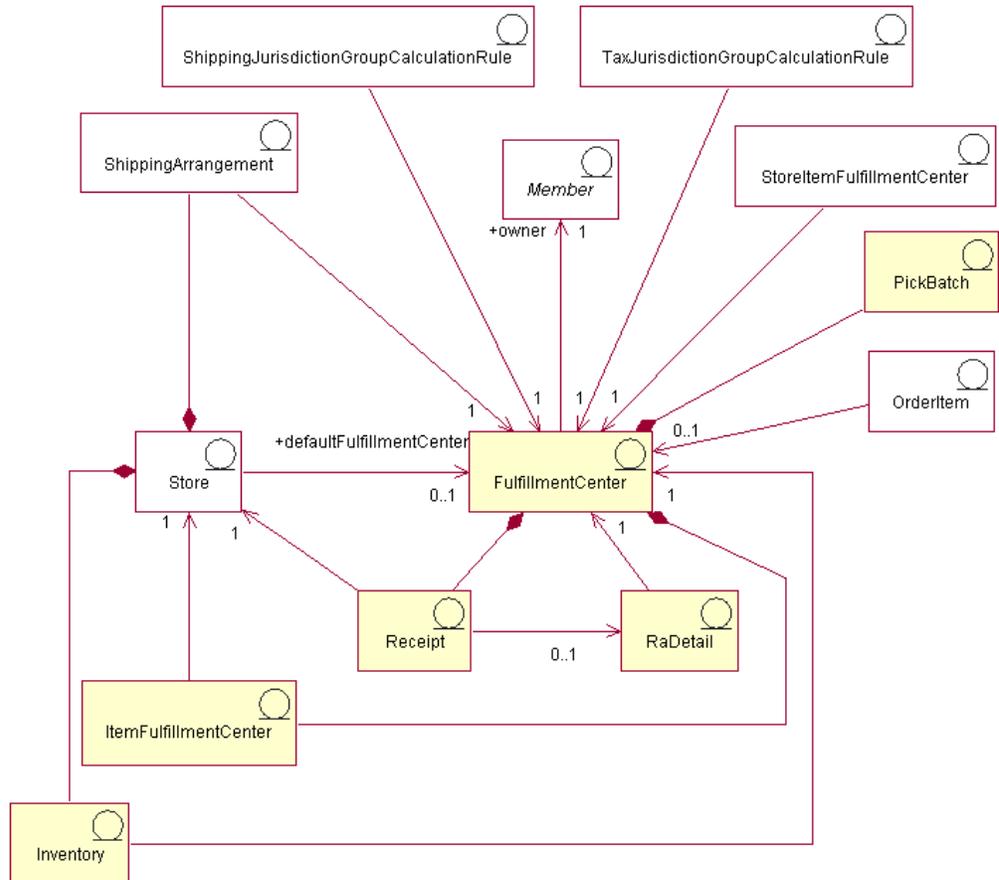
I prodotti sono configurati per l'adempimento ordini nella procedura guidata Prodotto e nel blocco appunti Prodotto. Il blocco appunti Prodotto include opzioni per tenere traccia dell'inventario, consentire sospesi, imporre sospesi, rilasciare i prodotti separatamente e specificare che il prodotto non può essere restituito.

Normalmente vi è un numero di persone che lavorano contemporaneamente in un centro di evasione ordini, ciascuna con una differente attività da eseguire. WebSphere Commerce Accelerator divide le attività più comuni in ruoli, che vengono poi assegnati all'utente. In WebSphere Commerce Accelerator, nel momento in cui ci si collega, è possibile selezionare un centro di evasione ordini se sono stati assegnati uno o più ruoli pertinenti alla gestione.

Nota: Per ulteriori informazioni sull'evasione ordini, sui centri di evasione ordini e sui ruoli, fare riferimento alla guida in linea di WebSphere Commerce.

Risorse evasione ordini in WebSphere Commerce

Per comprendere le risorse evasione ordini, è necessario conoscere le relazioni tra l'evasione ordini e il negozio. Tali relazioni possono essere spiegate utilizzando un modello di informazioni. Le sezioni di seguito riportate descrivono le relazioni tra un inventario, un negozio e altre risorse.



Questo grafico, insieme a tutti gli altri grafici della sezione relativa ai dati del negozio fa parte del modello di informazioni di WebSphere Commerce Server. Per ulteriori informazioni sul modello di informazioni, fare riferimento alla sezione "Modello di informazioni dati del negozio" a pagina 27. Per ulteriori informazioni sulle convenzioni utilizzate in questo grafico, fare riferimento all'Appendice A, "Legenda UML" a pagina 317.

Centro di evasione ordini

Nel grafico sopra riportato, il *centro di evasione ordini* è nel punto centrale del relativo processo. Un centro di evasione ordini ha un proprietario, definito nella tabella MEMBER. Ogni negozio è associato ad un centro di evasione ordini e dispone di un centro di evasione ordini predefinito. Ad un centro di evasione ordini possono essere associati diversi negozi. Esistono varie relazioni tra il negozio ed il centro di evasione ordini, come indicato nel grafico. Per ulteriori informazioni sulle risorse del negozio, fare riferimento alla sezione "Risorse del negozio in WebSphere Commerce" a pagina 43.

Ricevute

I centri di evasione ordini ricevono l'inventario degli articoli su base giornaliera, settimanale o mensile. Ricevuto l'inventario per un articolo, viene creata una *ricevuta* nella tabella RECEIPT in cui sono registrate le informazioni sulla quantità ricevuta e sul negozio che gestisce l'inventario. Man mano che gli ordini sono elaborati, la tabella RECEIPT viene aggiornata per riflettere i livelli di inventario disponibili. Per informazioni sulla creazione delle ricevute, fare riferimento alla sezione "Creazione di risorse inventario in WebSphere Commerce" a pagina 176.

RaDetail

RaDetail presenta informazioni dettagliate sugli articoli del record di un determinato inventario. Queste informazioni possono essere utilizzate per prevedere quando l'inventario verrà ricevuto da un centro di evasione ordini e per fornire ai clienti le date di spedizione previste per gli ordini in sospenso.

Inventario

Un negozio presenta un *inventario* che è associato al centro di evasione ordini. L'inventario comprende tutto ciò che può essere fisicamente considerato in un centro di evasione ordini. Ogni inventario è associato ad un unico negozio e ad un unico centro di evasione ordini. Le informazioni sull'inventario che un negozio gestisce al centro di evasione ordini sono registrate anche come quantità riservate, quantità in sospenso e quantità assegnate agli ordini in sospenso. Queste informazioni vengono memorizzate nella tabella ITEMFFMCTR. Per ulteriori informazioni sull'inventario e le risorse inventario, fare riferimento al Capitolo 21, "Risorse inventario" a pagina 173.

Disposizioni sulla spedizione

L'ultima relazione tra il negozio e il centro di evasione ordini riguarda le *disposizioni sulla spedizione*. Le disposizioni sulla spedizione indicano che un centro di evasione ordini può spedire i prodotti per conto di un negozio mediante una determinata modalità di spedizione. Ogni negozio ha disposizioni sulla spedizione con il centro di evasione ordini e viceversa. Le disposizioni sulla spedizione sono impostate nella tabella SHPARRANGE. Per informazioni sulla creazione delle disposizioni sulla spedizione, fare riferimento alla sezione "Creazione di risorse evasione ordini per la spedizione" a pagina 146.

Altre risorse evasione ordini

Esistono altre relazioni con il centro di evasione ordini che non sono direttamente collegate ad un negozio. Una raccolta articoli, ad esempio, è una relazione associata ad un centro di evasione ordini. Una raccolta articoli raggruppa i rilasci di ordini affinché siano elaborati come unità ad un centro di evasione ordini e crea note di raccolta o di accompagnamento. Una volta che gli articoli sono stati raccolti e confezionati, un rilascio di ordine può essere spedito e la spedizione può essere confermata. Le informazioni sulla raccolta articoli sono contenute nella tabella PICKBATCH. Ogni articolo dell'ordine è associato anche ad un centro di evasione ordini. Un articolo è una particolare istanza di un prodotto, definita dagli attributi. Le informazioni su ciascun articolo di un ordine sono contenute nella tabella ORDERITEM. Per ulteriori informazioni sulle risorse dell'ordine, fare riferimento al Capitolo 22, "Risorse ordini" a pagina 177.

Come altre entità, un centro di evasione ordini presenta delle regole che governano alcune delle sue azioni. Ciascun centro di evasione ordini presenta delle regole relative alle tasse e alle spese di spedizione. Tali regole sono definite

rispettivamente nelle tabelle TAXJCRULE e SHPJCRULE. Per ulteriori informazioni sulle risorse tasse e spedizione, fare riferimento al Capitolo 18, "Risorse spedizione" a pagina 135 e "Risorse tasse in WebSphere Commerce" a pagina 153.



Per informazioni più dettagliate sulla struttura delle risorse evasione ordini di WebSphere Commerce Server, fare riferimento ai modelli di dati di evasione ordini nella guida in linea di WebSphere Commerce.

Creazione di risorse evasione ordini in WebSphere Commerce

Perché il negozio possa spedire le merci ad un cliente, è necessario definire i centri di evasione ordini, o centri, che forniscono le merci. Creare queste informazioni in formato file XML che possono essere caricati nel database mediante il pacchetto Loader. Per ulteriori informazioni sul pacchetto Loader, consultare la Parte 7, "Pubblicazione del negozio" a pagina 213.

Per creare le risorse evasione ordini per il negozio mediante un file XML, procedere come segue:

1. Rivedere i file XML utilizzati per creare le risorse evasione ordini per i negozi di esempio. Tutti i file per i negozi di esempio si trovano nel file dell'archivio negozio corrispondente.

I file dell'archivio negozio si trovano nelle seguenti directory:

-  NT unità:\WebSphere\CommerceServer\samplestores
-  2000 unità:\Program Files\WebSphere\CommerceServer\samplestores
-  AIX /usr/WebSphere/CommerceServer/samplestores
-  Solaris /opt/WebSphere/CommerceServer/samplestores
-  Linux /opt/WebSphere/CommerceServer/samplestores
-  400 /qibm/proddata/WebCommerce/samplestores

Nota: La guida in linea di WebSphere Commerce contiene le informazioni su ciascuna delle risorse dati contenute nei negozi di esempio.

Ogni negozio di esempio presenta un file `fulfillment.xml`, che comprende informazioni sull'evasione ordini. Per visualizzare il file `fulfillment.xml` nell'archivio negozio, decomprimere il file con un programma ZIP. Il file `fulfillment.xml` si trova nella directory dei dati.

2. Fare riferimento alle informazioni riportate in Appendice B, "Creazione di dati" a pagina 319.
3. Creare un file `fulfillment.xml`, copiando uno dei file `fulfillment.xml` presenti negli archivi negozio di esempio o creandone uno nuovo. Per ulteriori informazioni, fare riferimento al file DTD corrispondente a `fulfillment.xml`. I file DTD si trovano nelle seguenti directory:

-  NT unità:\WebSphere\CommerceServer\xml\sar
-  2000 unità:\Program Files\WebSphere\CommerceServer\xml\sar
-  AIX /usr/WebSphere/CommerceServer/xml/sar
-  Solaris /opt/WebSphere/CommerceServer/xml/sar
-  Linux /opt/WebSphere/CommerceServer/xml/sar
-  400 /qibm/proddata/WebCommerce/xml/sar

4. Definire il centro o i centri di evasione ordini supportati dal negozio:
- a. Utilizzando il seguente esempio come guida, definire un centro di evasione ordini nel file XML per la tabella FFMCENTER:

```
<ffmcenter
  ffmcenter_id="@ffmcenter_id_1"
  member_id="&MEMBER_ID"
  name="ToolTech Home"
  defaultshipoffset="0"
  markfordelete="0"
/>
```

dove

- ffmcenter_id è una chiave univoca generata.
 - member_id è il proprietario del centro di evasione ordini.
 - name è una stringa che, insieme al proprietario, identifica il centro di evasione ordini in modo univoco.
 - defaultshipoffset è una stima del numero di secondi necessari per la spedizione di un articolo da questo centro di evasione ordini. Questo valore può essere sovrascritto nella tabella STORITMFFC.
 - markfordelete indica se il centro di evasione ordini deve essere cancellato, così come segue: 0 = non cancellare. 1 = cancellare se non più in uso. Per ulteriori dettagli, consultare le informazioni sul programma di utilità Cleanup del database nella guida in linea di WebSphere Commerce.
- b. Utilizzando il seguente esempio come guida, descrivere un centro di evasione ordini nel file XML per la tabella FFMCENTDS. Se il negozio da creare è multiculturale, è necessario includere queste informazioni in un file XML specifico delle locale.

```
<ffmcentds
  ffmcenter_id="@ffmcenter_id_1"
  description="The fulfillment center that supplies products to ToolTech."
  language_id="&en_US"
  displayname="ToolTech Fulfillment"
  staddress_id="@staddress_id_en_US_1"
/>
```

dove

- ffmcenter_id è una chiave univoca generata.
 - description è una descrizione del centro di evasione ordini, che può essere visualizzata da un cliente.
 - language_id è la lingua in cui vengono visualizzate queste informazioni.
 - displayname è il nome del centro di evasione ordini, che può essere visualizzato da un cliente.
 - staddress_id è l'ubicazione fisica del centro di evasione ordini.
- c. Ripetere i passi a e b per tutti i centri di evasione ordini supportati dal negozio.



Per ulteriori informazioni sull'uso di @ e &, fare riferimento all'Appendice B, "Creazione di dati" a pagina 319.

Creazione di risorse evasione ordini del negozio

Una volta definito il centro o i centri di evasione ordini che forniranno le merci al negozio, è necessario associare un centro a ciascun prodotto venduto. Ciò significa che è necessario identificare il centro di evasione ordini che fornisce ciascun prodotto. Per creare questa relazione, aggiungere informazioni alla tabella INVENTORY. Creare queste informazioni in formato file XML che possono essere caricati nel database mediante il pacchetto Loader. Per ulteriori informazioni sul pacchetto Loader, consultare la Parte 7, "Pubblicazione del negozio" a pagina 213.

Nota:

1. Per poter associare un negozio ad un centro di evasione ordini, è necessario creare le risorse del negozio. Per ulteriori informazioni sulla creazione delle risorse del negozio, fare riferimento alla sezione "Creazione di risorse dati del negozio in un file XML" a pagina 45. Per creare le risorse evasione ordini del negozio, è necessario anche creare le risorse catalogo. Per ulteriori informazioni, consultare la sezione "Visualizzazione delle risorse catalogo del negozio" a pagina 77.
2. Creare risorse evasione ordini del negozio solo se si desidera implementare un'evasione ordini non ATP. La tabella INVENTORY non viene utilizzata da un negozio che includa funzioni ATP.

Per creare la relazione di evasione ordini per il negozio mediante un file XML, procedere come segue:

1. Rivedere i file XML utilizzati per creare le risorse evasione ordini del negozio nei negozi di esempio. Tutti i file per i negozi di esempio si trovano nel file dell'archivio negozio corrispondente.

I file dell'archivio negozio si trovano nelle seguenti directory:

-  NT unità:\WebSphere\CommerceServer\samplestores
-  2000 unità:\Program Files\WebSphere\CommerceServer\samplestores
-  AIX /usr/WebSphere/CommerceServer/samplestores
-  Solaris /opt/WebSphere/CommerceServer/samplestores
-  Linux /opt/WebSphere/CommerceServer/samplestores
-  400 /qibm/proddata/WebCommerce/samplestores

Nota: La guida in linea di WebSphere Commerce contiene le informazioni su ciascuna delle risorse dati contenute nei negozi di esempio.

Ogni negozio di esempio presenta un file `storefulfill.xml`, che comprende informazioni sull'evasione ordini del negozio. Per visualizzare il file `storefulfill.xml` nell'archivio negozio, decomprimere il file con un programma ZIP. Il file `storefulfill.xml` si trova nella directory dei dati.

2. Fare riferimento alle informazioni riportate in Appendice B, "Creazione di dati" a pagina 319.
3. Creare un file `storefulfill.xml`, copiando uno dei file `storefulfill.xml` presenti negli archivi negozio di esempio o creandone uno nuovo. Per ulteriori informazioni, fare riferimento al file DTD corrispondente a `storefulfill.xml`. I file DTD si trovano nelle seguenti directory:

-  NT unità:\WebSphere\CommerceServer\xml\sar
-  2000 unità:\Program Files\WebSphere\CommercServer\xml\sar

-  /usr/WebSphere/CommerceServer/xml/sar
 -  /opt/WebSphere/CommerceServer/xml/sar
 -  /opt/WebSphere/CommerceServer/xml/sar
 -  /qibm/proddata/WebCommerce/xml/sar
4. Utilizzando il seguente esempio come guida, nel file XML creare una relazione del centro di evasione ordini del negozio aggiungendo informazioni alla tabella INVENTORY.

```
<inventory
catentry_id="@catentry_id_1470"
quantity="100"
ffmcenter_id="@ffmcenter_id_1"
store_id="@storeent_id_1"
quantitymeasure="C62"
inventoryflags="0"
/>
```

dove

- `catentry_id` è la voce di catalogo fornita da questo centro di evasione ordini.
 - `quantity` è la quantità, in unità indicate da `QUANTITYMEASURE`, disponibile in questo centro di evasione ordini.
 - `ffmcenter_id` è il centro di evasione ordini che fornisce l'inventario.
 - `store_id` è il negozio per il quale l'inventario viene fornito.
 - `quantitymeasure` è l'unità di misura per `QUANTITY`.
 - `inventoryflags` sono gli indicatori per `QUANTITY` utilizzato.
 - 1 = noUpdate. Il comando di attività `UpdateInventory` predefinito non aggiorna `QUANTITY`.
 - 2 = noCheck. I comandi di attività `CheckInventory` e `UpdateInventory` predefiniti non eseguono il controllo di `QUANTITY`.
5. Ripetere il passo 3 per ciascuna voce di catalogo presente nel negozio.



Per ulteriori informazioni sull'uso di @ e &, fare riferimento all'Appendice B, "Creazione di dati" a pagina 319.

Capitolo 12. Risorse Campagne

Le campagne vengono utilizzate per organizzare le attività di marketing. Di solito, le campagne vengono promosse da un responsabile del marketing o delle attività promozionali. Spesso sono associate ad una serie di obiettivi. Ad esempio, una campagna "Ritorno a scuola" può avere l'obiettivo di incrementare le vendite di abbigliamento per bambini durante la campagna.

Campagne in WebSphere Commerce

All'interno di WebSphere Commerce, le campagne contengono i numeri delle iniziative delle campagne, che definiscono una condizione. Le iniziative della campagna generano contenuti destinati ai clienti, quando la condizione definita viene valutata per essere vera. Il risultato è che una campagna è l'elemento di marketing di livello maggiore che organizza le iniziative.

Le iniziative della campagna possono essere associate ad una campagna che contiene una collezione di iniziative. Ad esempio, se un negozio di forniture per ufficio presenta la campagna "Ritorno a scuola", le iniziative saranno responsabili delle attività di livello inferiore, come la promozione degli sconti su penne e quaderni per tutti i clienti registrati ed elencati come studenti nel campo Occupazione.

Le iniziative della campagna sono in grado di visualizzare tre tipi di contenuti dinamici:

- Iniziative di vendita suggestive
- Raccomandazioni basate sul filtraggio collaborativo
- Campagne pubblicitarie

Il contenuto vendite è progettato per fornire raccomandazioni su categoria e prodotto in base a ruoli, destinati ad uno specifico pubblico di clienti, in base al profilo di un cliente e ad altri comportamenti del cliente. Le iniziative che visualizzano questo tipo di contenuto si utilizzano per creare opportunità di vendita incrociata e al rialzo.

Le raccomandazioni basate sul filtraggio collaborativo sono previste per creare raccomandazioni sul prodotto, ma utilizzano un diverso algoritmo di raccomandazione, che destina gli articoli in base al comportamento generale dei clienti, piuttosto che a regole predefinite.

Le campagne pubblicitarie sono progettate per fornire il contenuto pubblicitario destinato ad uno specifico pubblico di clienti, in base agli stessi criteri utilizzati per la vendita ma sono utilizzate per aumentare la pubblicità di un cliente circa le attività del negozio in linea, evidenziando le offerte speciali ed aumentando la pubblicità della marca.

Le iniziative possono essere incorporate in una qualsiasi pagina del sito. Quando viene progettato un sito, vengono visualizzati degli speciali segnaposti, chiamati spot di e-Marketing. Quando un cliente visita il sito, questi segnaposti vengono riposizionati in base ai contenuti specifici. Le posizioni di destinazione vengono assegnate durante la pianificazione delle iniziative per essere visualizzate in spot di

e-Marketing nelle posizioni desiderate. Per ulteriori informazioni sull'aggiunta di spot e-Marketing al negozio, consultare Capitolo 32, "Aggiunta di spot di e-Marketing al negozio" a pagina 309.

Le iniziative di campagna contengono una condizione che determina quando e a chi devono essere visualizzate. Questa condizione viene definita quando viene creata l'iniziativa e può essere modificata durante la durata dell'iniziativa per regolare la visibilità dell'iniziativa ed il contenuto visualizzato.

Le iniziative consentono di formulare delle statistiche sul loro impiego. Queste statistiche possono essere visualizzate mediante WebSphere Commerce Accelerator da Venditori, Responsabili del marketing e Responsabili della commercializzazione. Le statistiche indicano la frequenza di consultazione di un'iniziativa per ciascun spot di e-Marketing in cui è implementata. Queste statistiche forniscono le indicazioni sull'efficacia dell'iniziativa e le percentuali comparate di successo tra le varie ubicazioni in cui viene visualizzata.

Creazione di risorse campagne in WebSphere Commerce

La campagne e le iniziative di campagna vengono in genere create da un responsabile del marketing o delle attività promozionali mediante le wizard Campagna e Iniziativa campagna in WebSphere Commerce Accelerator. Per ulteriori informazioni, fare riferimento alla guida in linea di WebSphere Commerce.

Per ulteriori informazioni sull'aggiunta di spot e-Marketing al negozio, consultare Capitolo 32, "Aggiunta di spot di e-Marketing al negozio" a pagina 309.

Capitolo 13. Risorse pagamento

WebSphere Commerce supporta IBM Payment Manager. Per creare le risorse pagamento per il negozio, specificare se il negozio utilizza Payment Manager e, in questo caso, quale tipo di cassetta di pagamento e marca accetta il negozio.

Per specificare queste informazioni, attenersi alla procedura di seguito riportata:

- Creare dati sul pagamento in formato di file XML (`paymentinfo.xml`) caricati durante la pubblicazione del negozio mediante Servizi del negozio. Payment Manager viene configurato con il tipo di venditore e marca specificati per il negozio da pubblicare. Per ulteriori informazioni, fare riferimento alla sezione "Creazione di risorse pagamento mediante un file XML".

Nota: Il file `paymentinfo.xml` non compila le tabelle del database di WebSphere Commerce Server, ma configura Payment Manager. Il file `paymentinfo.xml` è applicabile solo se si utilizza una carta di credito offline come metodo di pagamento. Per configurare altri metodi di pagamento, fare riferimento al punto successivo.

- Completare l'impostazione di Payment Manager per il proprio negozio utilizzando la Console di gestione oppure l'interfaccia utente di Payment Manager. Se si utilizza la Console di gestione, le voci del menu appariranno nel menu Payment Manager. Se si utilizza l'interfaccia utente di Payment Manager, le voci di menu saranno visualizzate sotto Gestione nel frame di navigazione. Per ulteriori informazioni, fare riferimento alla sezione della guida in linea di WebSphere Commerce "Impostazione di Payment Manager per il negozio."

Creazione di risorse pagamento mediante un file XML

Per creare le risorse pagamento per il negozio mediante un file XML, procedere come segue:

1. Rivedere i file XML utilizzati per creare le risorse pagamento nei negozi di esempio. Tutti i file per i negozi di esempio si trovano nel file dell'archivio negozio corrispondente.

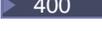
I file degli archivi negozio si trovano nelle seguenti directory:

-  NT unità:\WebSphere\CommerceServer\samplestores
-  2000 unità:\Program Files\WebSphere\CommerceServer\samplestores
-  AIX /usr/WebSphere/CommerceServer/samplestores
-  Solaris /opt/WebSphere/CommerceServer/samplestores
-  Linux /opt/WebSphere/CommerceServer/samplestores
-  400 /qibm/proddata/WebCommerce/samplestores

Nota: La guida in linea di WebSphere Commerce contiene le informazioni su ciascuna delle risorse dati contenute nei negozi di esempio.

Ciascun negozio di esempio presenta un file `paymentinfo.xml`, contenente le informazioni sul pagamento. Per visualizzare il file `paymentinfo.xml` nell'archivio negozio, decomprimere il file con un programma ZIP. I file `paymentinfo.xml` si trovano nella directory dei dati.

2. Creare un file `paymentinfo.xml`, copiando uno dei file `paymentinfo.xml` presenti negli archivi negozio di esempio o creandone uno nuovo. Per ulteriori informazioni, fare riferimento al file DTD corrispondente a `paymentinfo.xml`. I file DTD si trovano nelle seguenti directory:

-  unità:\WebSphere\CommerceServer\xml\sar
-  unità:\Program Files\WebSphere\CommerceServer\xml\sar
-  /usr/WebSphere/CommerceServer/xml/sar
-  /opt/WebSphere/CommerceServer/xml/sar
-  /opt/WebSphere/CommerceServer/xml/sar
-  /qibm/proddata/WebCommerce/xml/sar

3. Abilitare o disabilitare Payment Manager.

a. Utilizzando il seguente esempio come guida, nel file XML abilitare o disabilitare Payment Manager e specificare i tipi di cassette di pagamento, valute e marche accettati dal negozio

```
<paymentinfo>
  <PaymentManager enable="yes"/>
  <Cassette type="OfflineCard">
    <Account currency="USD">
      <Brand type="MasterCard"/>
      <Brand type="VISA"/>
      <Brand type="American Express"/>
    </Account>
    <Account currency="EUR">
      <Brand type="MasterCard"/>
      <Brand type="VISA"/>
      <Brand type="American Express"/>
    </Account>
  </Cassette>
</paymentinfo>
```

dove:

- `enable` indica se Payment Manager è abilitato o disabilitato.
- `Cassette type` è il tipo di cassetta supportato.
- `Account currency` è la valuta supportata dal negozio. Il valore `Account currency` è necessario se si utilizza il tipo di cassetta `OfflineCard`. La valuta deve essere identificata in un codice a tre lettere conforme allo standard ISO 4217. Ad esempio, "USD" per dollari U.S.A.
- `Brand type` è il tipo di carta di credito supportato da account e valuta.

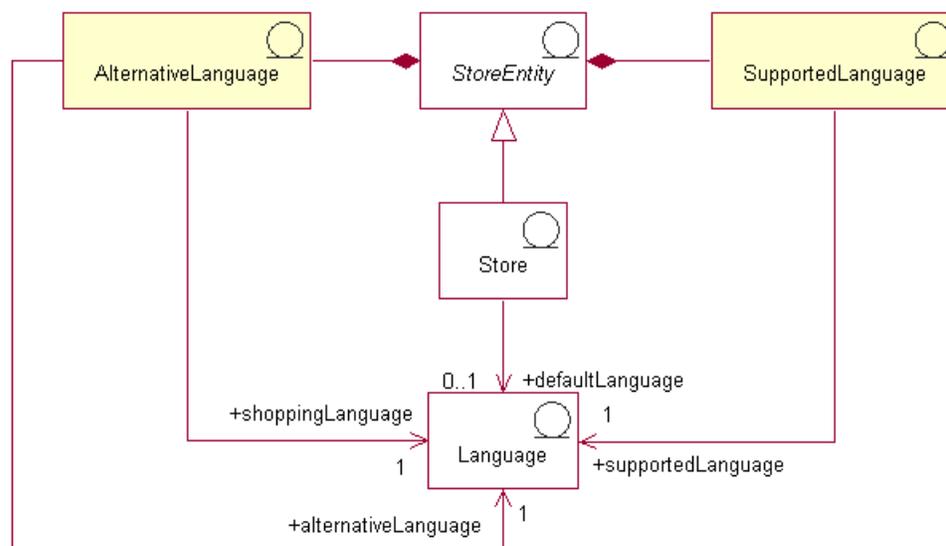
Capitolo 14. Risorse lingua

In WebSphere Commerce, in un sito possono essere definite più lingue utilizzate. La tabella LANGUAGE definisce dieci lingue supportate che includono il tedesco, il cinese tradizionale e semplificato, il giapponese, il coreano, l'italiano, il francese, lo spagnolo, il portoghese brasiliano e l'inglese. Nei siti possono essere definite altre lingue o dialetti delle lingue esistenti, per predisporre il modo in cui le informazioni vengono presentate ai clienti di diverse culture o aree demografiche.

Risorse lingua in WebSphere Commerce

Per comprendere le risorse lingua è necessario conoscere le relazioni tra le lingue e il negozio. Tali relazioni possono essere spiegate utilizzando il modello di informazioni di seguito riportato. Questa sezione descrive le relazioni e le associazioni tra una lingua, un negozio e altre risorse.

Il grafico di seguito riportato illustra il modello di informazioni per la risorsa lingua.



Esistono quattro classificazioni di lingue in WebSphere Commerce, lingue predefinite, lingue supportate, lingue alternative e lingue di shopping. Ciascuna di queste classificazioni ha nel negozio un ruolo diverso. Tutte le lingue sono presenti nella tabella LANGUAGE.

Lingua predefinita

A ciascun negozio è associata una *lingua predefinita*. Si tratta della lingua che il negozio ha scelto di utilizzare come lingua principale, che verrà visualizzata ai clienti che non scelgono esplicitamente una lingua per gli acquisti. La lingua predefinita per un negozio è implicitamente supportata dal negozio stesso; vale a dire che il negozio deve sempre essere in grado di visualizzare le informazioni nella lingua predefinita o in una delle lingue alternative, se definite nella tabella

LANGPAIR. Quando le informazioni non sono disponibili in una delle lingue supportate o in una lingua alternativa, vengono visualizzate nella lingua predefinita.

Lingua supportata

Nella tabella STORELANG sono indicate le lingue supportate dal negozio. Un negozio deve essere sempre in grado di visualizzare le informazioni nelle *lingue supportate* o in una delle lingue alternative, se definite nella tabella LANGPAIR. Un negozio supporta anche tutte le lingue supportate dal relativo gruppo di negozi.

Lingua alternativa

Quando le informazioni non sono disponibili in una delle lingue supportate, il negozio tenta di visualizzare le informazioni in una *lingua alternativa*, se disponibile. Un negozio può specificare la sequenza in cui tentare le lingue alternative. Le lingue alternative per un negozio includono le lingue alternative per il relativo gruppo di negozi. Le lingue alternative possono essere utili quando alcune informazioni sono disponibili in un'unica lingua e devono essere rese disponibili ai clienti che fanno acquisti in una lingua diversa. Ciò potrebbe verificarsi, ad esempio, quando non tutte le informazioni sono state ancora tradotte in tutte le lingue supportate o quando sono supportati due dialetti molto simili della stessa lingua, talvolta con informazioni identiche.



Per informazioni più dettagliate sulla struttura delle risorse lingua di WebSphere Commerce Server, fare riferimento ai modelli di oggetti e dati della lingua nella guida in linea di WebSphere Commerce.

Creazione di risorse lingua in WebSphere Commerce

E' possibile definire le lingue supportate dal negozio in uno dei seguenti modi:

- Utilizzando gli strumenti Servizi del negozio
- In un file XML che verrà caricato dal Pacchetto Loader o mediante lo strumento di pubblicazione di Servizi del negozio

Nota: Gli strumenti Servizi del negozio utilizzano file XML precostituiti sotto forma di un archivio negozio.

Per ulteriori informazioni sulla definizione delle lingue supportate dal negozio mediante Servizi del negozio, fare riferimento alla guida in linea di WebSphere Commerce. Per ulteriori informazioni sulla definizione delle lingue supportate dal negozio in un file XML, fare riferimento alla sezione "Creazione di risorse dati del negozio in un file XML" a pagina 45.

Capitolo 15. Risorse valuta

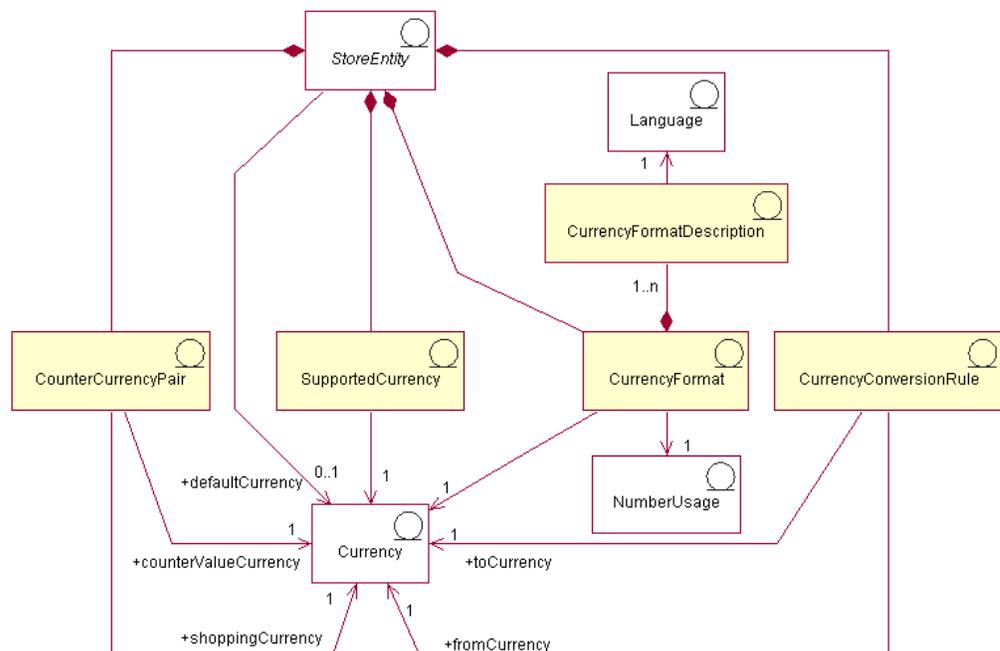
E' possibile visualizzare i prezzi nel sito in una determinata valuta o utilizzare più valute seguendo le istruzioni fornite per l'Euro. Per un sito che contiene più negozi, è possibile utilizzare valute diverse per i negozi o assegnare valute comuni all'intero gruppo di negozi. A seconda della natura del sito da creare, è possibile specificare quali valute utilizzare e come visualizzarle.

In WebSphere Commerce, è possibile consentire ai clienti di selezionare una valuta d'acquisto. La valuta di acquisto è la valuta con la quale i clienti pagano i prodotti in un negozio specifico. Tutti gli importi monetari nelle pagine di negozio vengono visualizzate in questa valuta. Se i clienti cambiano la valuta d'acquisto, i prezzi degli articoli aggiunti al carrello degli acquisti e i totali degli ordini saranno automaticamente convertiti, ricalcolati e visualizzati nella nuova valuta d'acquisto.

I clienti possono fare shopping in diverse valute, compreso l'Euro. L'Euro diventerà la valuta legale dell'Unione Europea il 1 Gennaio 1999 ed attualmente viene utilizzata nei mercati finanziari. I tassi di conversione tra Euro e le valute di tutti i paesi partecipanti sono fissi. Le banconote e le monete per la futura moneta unica dell'Unione Europea diventeranno disponibili dal 1° gennaio 2002. Sei mesi dopo, le valute nazionali esistenti saranno ritirate dalla circolazione definitivamente. Durante il periodo di transizione, tra il 1999 e il 2001, i venditori devono accettare sia le valute nazionali che gli Euro.

Risorse valuta in WebSphere Commerce

Il grafico di seguito riportato illustra la struttura della valuta in WebSphere Commerce Server:





Questo grafico, insieme a tutti gli altri grafici della sezione relativa ai dati del negozio fa parte del modello di informazioni di WebSphere Commerce Server. Per ulteriori informazioni sul modello di informazioni, fare riferimento alla sezione "Modello di informazioni dati del negozio" a pagina 27. Per ulteriori informazioni sulle convenzioni utilizzate in questo grafico, fare riferimento all'Appendice A, "Legenda UML" a pagina 317.

Nel grafico sopra riportato, la valuta è al centro del modello di informazioni. Ogni negozio, o gruppo di negozi, ha una valuta predefinita.

Formato valuta

Un'entità negozio può avere più regole di *formattazione valuta*. Se un negozio non dispone di una regola di formattazione per una particolare valuta, utilizza la regola di formattazione del relativo gruppo di negozi. I formati valuta vengono impostati nella tabella CURFORMAT.

Utilizzo numero

Ogni regola di valuta supportata è associata a un *utilizzo numero*. I numeri che rappresentano quantità e importi monetari possono essere arrotondati e formattati diversamente a seconda degli utilizzi associati. I negozi possono specificare regole di arrotondamento e formattazione diverse per i numeri visualizzati, in base al modo in cui vengono utilizzati; ad esempio, un negozio può arrotondare i prezzi delle unità a quattro cifre decimali specificando l'utilizzo prezzo unità ma arrotondare altri importi a due cifre decimali specificando l'utilizzo predefinito. L'utilizzo numero viene memorizzato nella tabella NUMBRUSG.

Descrizione del formato valuta

Una regola di formattazione della valuta può avere varie *descrizioni del formato valuta*. Una descrizione del formato della valuta illustra il modo in cui formattare (per motivi di visualizzazione) una determinata quantità monetaria in una particolare valuta e lingua. Ogni descrizione è associata ad una lingua nella tabella LANGUAGE. Per ulteriori informazioni sulle risorse lingua, fare riferimento al Capitolo 14, "Risorse lingua" a pagina 121. Le descrizioni dei formati valuta sono memorizzate nella tabella CURFMTDESC.

Valuta supportata

Un'entità negozio può avere varie *valute supportate*. Una valuta supportata è la valuta in cui è accettato il pagamento.

Regole di conversione valute

Tutte le valute presentano regole che governano la conversione verso e da altre valute. Ogni *regola di conversione valute* può essere utilizzata per convertire un prezzo (memorizzato nel database in una particolare valuta) in un importo che il cliente dovrà pagare in una valuta supportata.

Valuta di controvalore

Le *valute di controvalore* sono gli importi di valuta visualizzati con una valuta supportata. Non possono essere utilizzati per gli acquisti ma solo per scopi informativi. Se i clienti decidono di fare shopping in Euro, possono visualizzare nel negozio l'importo monetario dell'unione monetaria europea e altri importi di valuta. Gli importi di valuta vengono convertiti in tutti i controvalori. I controvalori della valuta sono accoppiati ad una valuta supportata, ad esempio il

fiorino dei Paesi Bassi e l'Euro. Le coppie di controvalori sono memorizzate nella tabella CURCVLIST.



Per informazioni più dettagliate sulla struttura delle risorse valuta in WebSphere Commerce Server, fare riferimento al modello di dati della valuta nella guida in linea di WebSphere Commerce.

Creazione di risorse valuta in WebSphere Commerce

Gli strumenti Servizi del negozio di WebSphere Commerce consentono di aggiungere le valute supportate al negozio e di selezionare una valuta predefinita per il negozio: Per ulteriori informazioni sulle risorse che è possibile modificare con gli strumenti Servizi del negozio, fare riferimento alla sezione "Modifica delle risorse database del negozio" della guida in linea di WebSphere Commerce.

Nota: Gli strumenti Servizi del negozio utilizzano file XML precostituiti sotto forma di un archivio negozio.

E' possibile aggiungere al negozio le valute supportate e una valuta corrente mediante un file XML che può essere caricato nel database mediante il pacchetto Loader. Questo metodo consente anche di creare altri tipi di risorse valuta, inclusi i tassi di conversione della valuta e i controvalori della valuta corrente.

Per informazioni sulla modifica delle risorse valuta di un archivio negozio esistente, fare riferimento alla guida in linea di WebSphere Commerce. Per informazioni sulla creazione di nuove risorse valuta sotto forma di file XML, fare riferimento alla sezione "Creazione di risorse valuta mediante un file XML".

Creazione di risorse valuta mediante un file XML

Creare le risorse valuta per il negozio in formato file XML che possono essere caricati nel database mediante il pacchetto Loader. Per ulteriori informazioni sul pacchetto Loader, consultare la Parte 7, "Pubblicazione del negozio" a pagina 213.

Per creare le risorse valuta per il negozio mediante un file XML, procedere come segue:

1. Rivedere i file XML utilizzati per creare risorse valuta nei negozi di esempio. Tutti i file per i negozi di esempio si trovano nel file dell'archivio negozio corrispondente.

I file dell'archivio negozio si trovano nelle seguenti directory:

-  NT unità:\WebSphere\CommerceServer\samplstores
-  2000 unità:\Program Files\WebSphere\CommerceServer\samplstores
-  AIX /usr/WebSphere/CommerceServer/samplstores
-  Solaris /opt/WebSphere/CommerceServer/samplstores
-  Linux /opt/WebSphere/CommerceServer/samplstores
-  400 /qibm/proddata/WebCommerce/samplstores

Nota: La guida in linea di WebSphere Commerce contiene le informazioni su ciascuna delle risorse dati contenute nei negozi di esempio.

Ogni negozio di esempio presenta un file `currency.xml`, che comprende informazioni sulla valuta. Per visualizzare i file `currency.xml` nell'archivio negozio, decomprimere il file con un programma ZIP. I file `currency.xml` si trovano nella directory dei dati.

2. Fare riferimento alle informazioni nell'Appendice B, "Creazione di dati" a pagina 319.
3. Creare un file `currency.xml`, copiando uno dei file `currency.xml` presenti negli archivi negozio di esempio o creandone uno nuovo. Per ulteriori informazioni, fare riferimento al file DTD corrispondente a `currency.xml`. I file DTD si trovano nelle seguenti directory:

-  NT unità:\WebSphere\CommerceServer\xml\sar
-  2000 unità:\Program Files\WebSphere\CommerceServer\xml\sar
-  AIX /usr/WebSphere/CommerceServer/xml/sar
-  Solaris /opt/WebSphere/CommercServer/xml/sar
-  Linux /opt/WebSphere/CommercServer/xml/sar
-  400 /qibm/proddata/WebCommerce/xml/sar

4. Definire le valute supportate dal negozio.
 - a. Utilizzando il seguente esempio come guida, definire le valute supportate dal negozio nel file XML per la tabella CURLIST:

```
<curlist currstr="USD" storeent_id="@storeent_id_1" />
```

dove:

- `currstr` è il codice di valuta a 3 caratteri ISO 4217 che rappresenta la valuta supportata. Tale codice viene visualizzato nella colonna SETCCURR della tabella SETCURR. Un negozio deve essere in grado di accettare pagamenti in tutte le valute supportate.
 - `storeent_id` è l'entità negozio.
- b. Ripetere per ciascuna valuta supportata dal negozio.



La valuta predefinita per il negozio è specificata nella tabella STOREENT. Per ulteriori informazioni, consultare la sezione "Creazione di risorse dati del negozio in un file XML" a pagina 45.

5. (Facoltativo) Il modo in cui vengono visualizzati i prezzi nel negozio dipende da come tali prezzi sono impostati. E' possibile definire i prezzi per ciascuna valuta utilizzata nel negozio, oppure solo per la valuta predefinita. Per ulteriori informazioni sull'impostazione dei prezzi, fare riferimento alla sezione "Creazione di risorse assegnazione prezzi in WebSphere Commerce" a pagina 92.

Se durante l'impostazione vengono definiti i prezzi solo per la valuta predefinita e si desidera visualizzare i prezzi del negozio in altre valute supportate, è necessario aggiungere i tassi di conversione al negozio. Utilizzare questi tassi di conversione per eseguire la conversione dalla valuta predefinita alla valuta supportata.

- a. Stabilire la valuta da cui eseguire la conversione, ad esempio il dollaro americano (USD) e la valuta o le valute in cui eseguire la conversione, ad esempio lo Yen (JPY). Per stabilire i codici di valuta ISO per ciascuna valuta, fare riferimento ai codici ISO 4217 relativi alle valute internazionali.
- b. Utilizzando il seguente esempio come guida, aggiungere informazioni sulla conversione alla tabella CURCONVERT:

```
<curconvert
storeent_id="@storeent_id_1"
fromcurr="USD"
tocurr="JPY"
factor="105.10"
multiplyordivide="M"
bidirectional="Y"
updatable="Y"
curconvert_id="@curconvert_id_1" />
```

dove:

- storeent_id è l'entità negozio.
- fromcurr è la valuta da cui si esegue la conversione. Generalmente, un importo nella valuta FROMCURR è parte di una regola o di altre informazioni utilizzate per determinare un prezzo, uno sconto, un costo di spedizione oppure importi simili associati ad un prodotto in vendita.
- tocurr è la valuta in cui si esegue la conversione. Generalmente, TOCURR è la valuta che il cliente desidera utilizzare per il pagamento. Gli importi in questa valuta fanno parte di un articolo dell'ordine, ad esempio, un prezzo unitario, un costo di spedizione o un importo di tassa.
- factor è il fattore di conversione.
- multiplyordivide è il valore di seguito riportato: Per eseguire la conversione da FROMCURR a TOCURR:
 - M = Moltiplica per FACTOR
 - D = Dividi per FACTOR

Per le regole bidirezionali, la conversione da TOCURR a FROMCURR è consentita utilizzando l'operazione inversa.

- bidirectional indica se la regola è bidirezionale o unidirezionale.
 - Y = bidirezionale
 - N = unidirezionale
 - updatable è un indicatore che deve essere utilizzato da un'interfaccia utente che gestisce le regole di conversione della valuta. I valori validi sono:
 - N = il tasso di conversione non può essere modificato
 - Y = il tasso di conversione può essere modificato
 - curconvert_id è una chiave univoca generata.
- c. Ripetere i passi a e b per tutte le valute in cui si desidera visualizzare i prezzi.



Anche se sono stati definiti i prezzi per tutte le valute supportate nelle informazioni sui prezzi, è possibile definire i tassi di conversione della valuta per le valute supportate nel negozio.

6. (Facoltativo) Se si desidera visualizzare i prezzi nella valuta di acquisto e nel relativo controvalore (ad esempio in fiorini olandesi e in euro), è necessario aggiungere le informazioni alla tabella CURCVLIST.
- a. Utilizzando il seguente esempio come guida, aggiungere informazioni sulla conversione alla tabella CURCVLIST:

```
<curcvlist
storeent_id="@storeent_id_1"
currstr="NLG"
countervaluecurr="EUR"
displayseq="1" />
```

dove:

- `storeent_id` è l'entità negozio.
- `currstr` sono i tre caratteri del codice valuta ISO 4217 che rappresentano la valuta. Questo codice, visualizzato nella colonna `SETCCURR` della tabella `SETCURR`, è la valuta da cui si esegue la conversione. Generalmente, un importo nella valuta `FROMCURR` è parte di una regola o di altre informazioni utilizzate per determinare un prezzo, uno sconto, un costo di spedizione oppure importi simili associati ad un prodotto in vendita.
- `countvaluecurr` sono i tre caratteri del codice valuta ISO 4217 che rappresentano la valuta di controvalore. Tale codice viene visualizzato nella colonna `SETCCURR` della tabella `SETCURR`.
- `displayseq` è il numero che indica l'ordine di presentazione della valuta di controvalore. Le valute di controvalore vengono visualizzate in ordine ascendente in base alla sequenza di visualizzazione del controvalore nella colonna `DISPLAYSEQ` della tabella `CURCVLIST`.



Per ulteriori informazioni sull'uso di @ e &, fare riferimento all'Appendice B, "Creazione di dati" a pagina 319.

Altre attività relative alla valuta

Per ulteriori informazioni sulla valuta in generale e su altre attività relative alla valuta, incluso:

- Aggiunta di nuove valute non supportate da WebSphere Commerce
- Modifica dei formati delle valute esistenti
- Aggiunta di nuove regole di formattazione per le nuove valute o per uno specifico negozio

fare riferimento alla guida in linea di WebSphere Commerce.

Capitolo 16. Risorse unità di misura

I prodotti possono essere venduti e l'inventario può essere tracciato in diverse unità di quantità, come chilogrammi, pollici, litri e così via. In base a queste unità, i prodotti possono essere ordinati in quantità minima e per multipli di quantità specifiche.

I comandi del controller utilizzano il valore UOM (unit of measure) per specificare l'unità di quantità. Se non viene specificato alcun parametro UOM, la quantità specificata dal cliente viene moltiplicata per la quantità nominale indicata nel catalogo nella tabella di database CATENTSHIP. Il risultato è noto come la quantità richiesta.

La quantità richiesta viene arrotondata per eccesso al multiplo della quantità più alta per la voce di catalogo. Ad esempio, se il multiplo è 2 chilogrammi e la quantità richiesta è 4,1 chilogrammi, il risultato dell'arrotondamento sarà 6 chilogrammi. La quantità arrotondata viene utilizzata durante il controllo dell'inventario. Se l'unità di quantità dell'inventario e l'unità di quantità della voce di catalogo sono differenti, è necessario che avvenga tra le due una conversione.

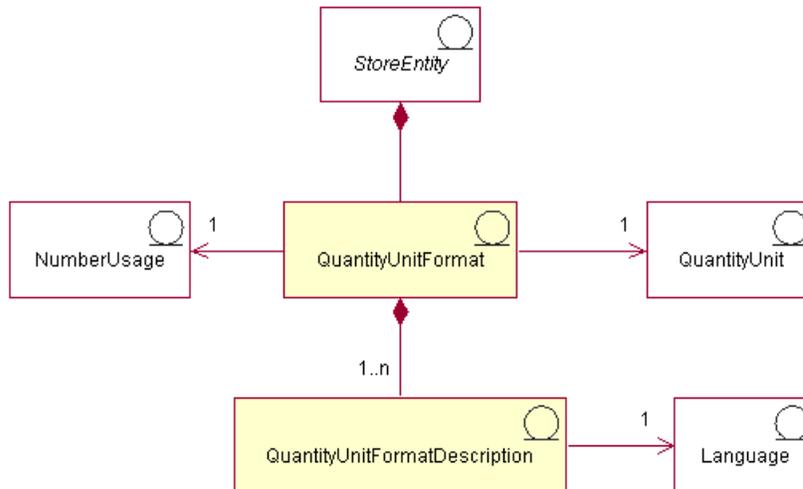
Quando è abilitato l'inventario ATP (Available to Promise) (fare riferimento alla colonna ALLOCATIONGOODFOR della tabella STORE), l'unità di quantità inventario viene definita nella colonna QUANTITYMEASURE della tabella BASEITEM. Diversamente, l'unità viene definita nella colonna QUANTITYMEASURE della tabella INVENTORY.

La quantità arrotondata divisa per la quantità nominale della voce di catalogo è nota come quantità normalizzata. La quantità normalizzata viene memorizzata nell'articolo dell'ordine o nell'articolo di interesse, a seconda del comando in esecuzione. Ad esempio, se la quantità arrotondata è 6 chilogrammi e la quantità nominale è 2 chilogrammi, la quantità normalizzata è 3.

Quando viene rilevata un'offerta per una voce di catalogo, la quantità richiesta può influire sull'offerta che propone il prezzo più vantaggioso, e quindi determina l'offerta da utilizzare. Ad esempio, se la quantità arrotondata è 6 kg e vi sono due offerte, una che indica un prezzo di 4 Euro per la quantità nominale di 2 chilogrammi ed una quantità minima di 10 chilogrammi e l'altra che specifica un prezzo di 4,50 Euro per la quantità nominale di 2 chilogrammi ed una quantità minima di 2 chilogrammi, è possibile utilizzare soltanto la seconda offerta.

Unità di misura in WebSphere Commerce

Il grafico di seguito riportato illustra la struttura delle unità di misura in WebSphere Commerce Server:



Questo grafico, insieme a tutti gli altri grafici della sezione relativa ai dati del negozio fa parte del modello di informazioni di WebSphere Commerce Server. Per ulteriori informazioni sul modello di informazioni, fare riferimento alla sezione "Modello di informazioni dati del negozio" a pagina 27. Per ulteriori informazioni sulle convenzioni utilizzate in questo grafico, fare riferimento all'Appendice A, "Legenda UML" a pagina 317.

Unità di quantità e formato dell'unità di quantità

Un'unità di quantità è l'unità di misura utilizzata nel negozio, ad esempio, chilogrammi, libbre, metri, pollici, litri e così via. Il formato dell'unità di quantità è il modo in cui l'unità di quantità è formattata nel negozio, ad esempio quante cifre decimali vengono utilizzate quando si visualizza l'unità di quantità.

Ciascun formato dell'unità di misura fa parte di una sola entità negozio, ma ciascuna entità negozio può avere diversi formati per le unità di quantità.

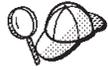
Un formato dell'unità di quantità può esserci per ciascuna unità di quantità e utilizzo numero e può avere una o più descrizioni, in base al numero di lingue supportate dal negozio.

Descrizione del formato dell'unità di quantità

La descrizione del formato dell'unità di quantità illustra il modo in cui formattare (per motivi di visualizzazione) una determinata quantità in una particolare unità di quantità, per una particolare lingua.

Utilizzo numero

L'utilizzo del numero definisce il modo in cui un numero viene utilizzato in un'applicazione. Ad esempio, adoperando i codici di utilizzo del numero nel codice WebSphere Commerce, è possibile scegliere il modo secondo il quale formattare o arrotondare quel numero (valuta o quantità). Questi codici (definiti nella tabella NUMBRUSG) consentono di formattare il numero in base alle regole specificate per quel tipo di utilizzo del numero nelle tabelle CURFORMAT, CURFMTDESC, QTYFORMAT e QTYFMTDESC. Ciò consente ai negozi di formattare i numeri in diversi modi per soddisfare i requisiti di una grande varietà di situazioni.



Per informazioni più dettagliate sulla struttura delle risorse unità di misura di WebSphere Commerce Server, fare riferimento al modello dati dell'unità di quantità nella guida in linea di WebSphere Commerce.

Creazione di unità di misura in WebSphere Commerce

Le unità di misura vengono stabilite nel database di WebSphere Commerce Server al momento della creazione di un'istanza. Per ulteriori informazioni, consultare Capitolo 5, "Risorse del sito" a pagina 39.

E' inoltre possibile definire in WebSphere Commerce nuove unità di misura da utilizzare nel negozio oppure cancellare le unità di misura che non si desidera utilizzare.

Per definire nuove unità di misura, aggiungere le informazioni alle seguenti tabelle di database:

- QTYUNIT
- QTUNITDSC
- QTYFORMAT
- QTYFMTDESC
- QTYUNITMAP
- QTYCONVERT

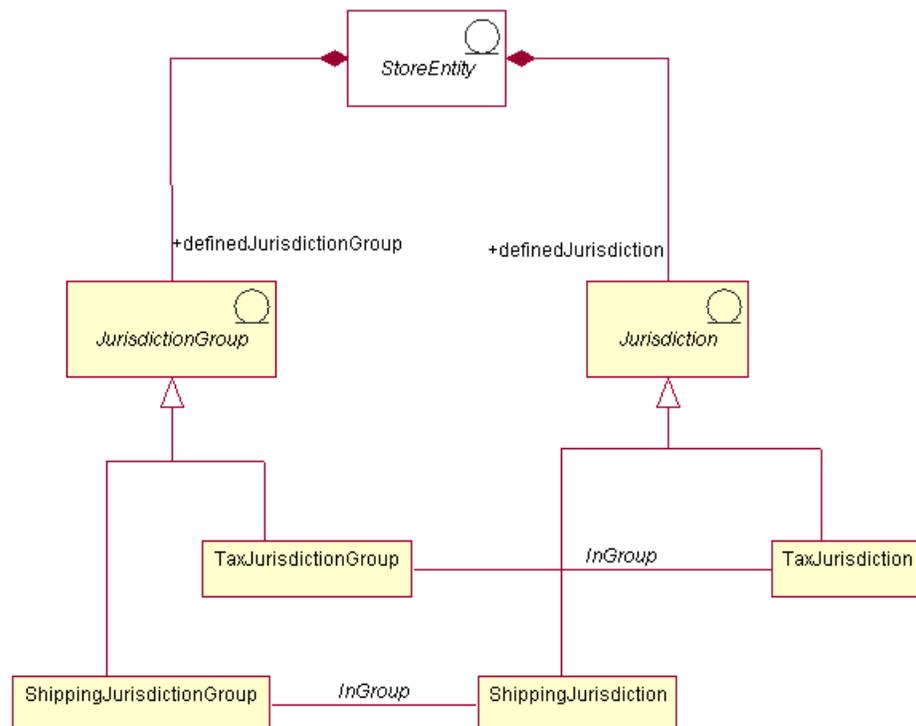
Capitolo 17. Risorse giurisdizioni

Le *giurisdizioni* sono aree o zone geografiche che rappresentano un paese o regione, una provincia o territorio oppure un codice postale a cui vengono vendute le merci. Le giurisdizioni sono raggruppate per formare *gruppi di giurisdizioni*.

I gruppi di giurisdizioni vengono utilizzati per il calcolo delle spese di spedizione e delle tasse sugli ordini. Ciò significa che un gruppo di giurisdizioni può essere utilizzato per qualificare le regole per il calcolo delle spese di spedizione e delle tasse utilizzate. Queste regole di calcolo qualificate sono applicabili agli articoli di un ordine solo se l'articolo viene spedito ad un indirizzo all'interno di una delle giurisdizioni del gruppo di giurisdizioni associato alla regola di calcolo. Quindi, le spese di spedizione e le tasse possono essere calcolate in modo diverso, in base agli indirizzi di spedizione per i diversi articoli dell'ordine.

Risorse giurisdizioni in WebSphere Commerce

Il grafico di seguito riportato illustra il modo in cui le giurisdizioni e i gruppi di giurisdizioni si inseriscono in WebSphere Commerce Server.



Questo grafico, insieme a tutti gli altri grafici della sezione relativa ai dati del negozio fa parte del modello di informazioni di WebSphere Commerce Server. Per ulteriori informazioni sul modello di informazioni, fare riferimento alla sezione "Modello di informazioni dati del negozio" a pagina 27. Per ulteriori informazioni sulle convenzioni utilizzate in questo grafico, fare riferimento all'Appendice A, "Legenda UML" a pagina 317.

In WebSphere Commerce, una giurisdizione o gruppo di giurisdizioni è parte di un negozio ed è esclusiva per il negozio o per il gruppo di negozi per il quale è stata creata. Ad esempio, se si creano tre giurisdizioni per il negozio e si cancella il negozio, vengono cancellate anche le giurisdizioni. Le giurisdizioni non possono essere utilizzate dagli altri negozi esistenti o dai negozi che verranno creati in futuro.

Tuttavia, se si creano giurisdizioni per un gruppo di negozi, le giurisdizioni non vengono cancellate quando vengono cancellati i negozi di quel gruppo. Le giurisdizioni sono disponibili per i nuovi negozi creati in quel gruppo di negozi.

WebSphere Commerce supporta due tipi di giurisdizioni: le giurisdizioni di spedizione e le giurisdizioni di tasse. Le giurisdizioni di spedizione sono raggruppate per formare gruppi di giurisdizioni di spedizione, che qualificano le regole per il calcolo delle spese di spedizione. Allo stesso modo, le giurisdizioni di tasse sono raggruppate per formare gruppi di giurisdizioni di tasse, che qualificano le regole per il calcolo delle tasse.



Per informazioni più dettagliate sulla struttura delle risorse giurisdizioni in WebSphere Commerce Server, fare riferimento al modello di dati delle giurisdizioni nella guida in linea di WebSphere Commerce.

Creazione di risorse giurisdizioni in WebSphere Commerce

Per applicare tasse e spese di spedizione, è necessario creare le risorse giurisdizioni per il negozio. Per ulteriori informazioni sulla creazione delle giurisdizioni, fare riferimento alle sezioni "Creazione di risorse tasse in WebSphere Commerce" a pagina 156 o "Creazione di risorse spedizione in WebSphere Commerce" a pagina 137.

Una volta create le giurisdizioni, è possibile modificarle o crearne di nuove, utilizzando i blocchi appunti Tasse e Spedizioni in Servizi del negozio.

Nota: Lo strumento Servizi del negozio crea automaticamente un gruppo di giurisdizioni per ogni giurisdizione creata. Servizi del negozio crea le giurisdizioni per i negozi ma non per i gruppi di negozi.

Nota: Servizi del negozio funziona solo sui dati precedentemente pubblicati sotto forma di un archivio negozio.

Capitolo 18. Risorse spedizione

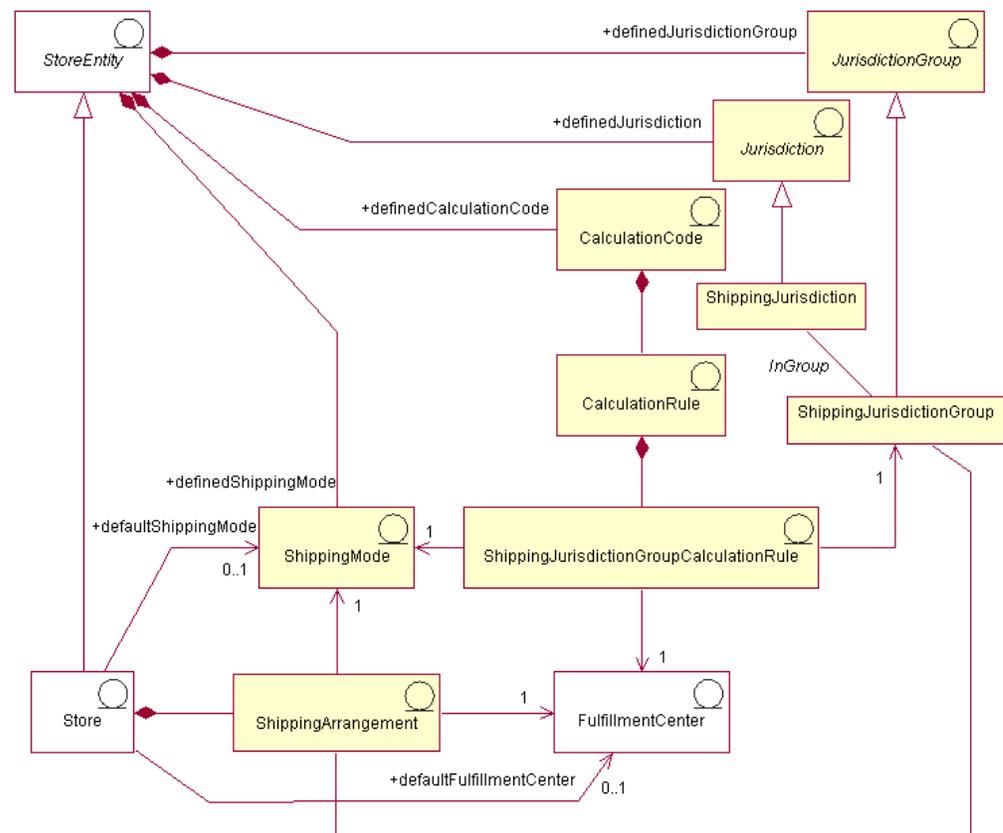
La spedizione è il modo in cui un negozio gestisce fisicamente la consegna delle merci ai clienti. Nella maggior parte dei casi, le merci vengono spedite da un centro di evasione ordini, un'agenzia separata responsabile dell'immagazzinamento della merce del negozio.

Per offrire servizi di spedizione e tariffe per tali servizi, un negozio creato con WebSphere Commerce deve includere quanto segue:

- Almeno una modalità di spedizione
- Almeno un codice di calcolo delle spedizioni
- Giurisdizioni e gruppi di giurisdizioni

Risorse spedizione in WebSphere Commerce

Il grafico di seguito riportato illustra la struttura della spedizione in WebSphere Commerce Server.



Questo grafico, insieme a tutti gli altri grafici della sezione relativa ai dati del negozio fa parte del modello di informazioni di WebSphere Commerce Server. Per ulteriori informazioni sul modello di informazioni, fare riferimento alla sezione "Modello di informazioni dati del negozio" a pagina 27. Per ulteriori informazioni sulle convenzioni utilizzate in questo grafico, fare riferimento all'Appendice A, "Legenda UML" a pagina 317.

Modalità di spedizione

La *modalità di spedizione* è il modo in cui vengono spedite le merci. Più specificamente, una modalità di spedizione è la combinazione di uno spedizioniere (che è una società che fornisce servizi di spedizione da un centro di evasione ordini ad un cliente) e di un servizio di spedizione offerto da quel corriere. Ad esempio, Società di spedizione ABC, servizio 24 ore su 24 e Società di spedizione ABC, consegna rapida sono modi di spedizione.

Un codice di calcolo è parte di un'entità negozio. Se l'entità negozio viene cancellata, vengono cancellate anche le modalità di spedizione definite all'interno dell'entità negozio. Un negozio può avere, facoltativamente, una modalità di spedizione predefinita.

Disposizioni sulla spedizione

Una *disposizione di spedizione* è una disposizione tra il negozio ed il centro di evasione ordini, che indica che il centro di evasione ordini spedisce merce per un particolare negozio mediante le modalità di spedizione specificate. In una disposizione di spedizione possono essere inserite alcune limitazioni, che includono il periodo di tempo in cui la disposizione di spedizione è valida e le giurisdizioni di spedizione.

Se la disposizione di spedizione è associata ad una modalità di spedizione, viene applicata solo a quella modalità di spedizione. In caso contrario, la disposizione di spedizione si applica a tutte le modalità di spedizione disponibili. Una disposizione di spedizione è parte di un negozio e viene cancellata se il negozio viene cancellato.

Codici di calcolo

I *codici di calcolo* vengono utilizzati per calcolare le spese di spedizione, vale a dire che il codice di calcolo di una spedizione indica il modo in cui le spese di spedizione vengono calcolate per gli articoli dell'ordine. Per calcolare le spese di spedizione per un articolo dell'ordine, è necessario assegnare i codici di calcolo della spedizione a una voce di catalogo o ad un gruppo di voci di catalogo.

Un codice di calcolo è parte di un'entità negozio. Un codice di calcolo può essere associato ad un'unica entità negozio, ma un negozio può avere diversi codici di calcolo. Se l'entità negozio viene cancellata, vengono cancellati anche i codici di calcolo associati all'entità negozio.



Per ulteriori informazioni sull'utilizzo dei codici di calcolo, consultare il manuale *IBM WebSphere Commerce Calculation Framework Guide*.

Regole di calcolo

Ciascun codice di calcolo presenta un insieme di *regole di calcolo*. Le spese di spedizione per un articolo dell'ordine possono variare in base alla modalità di spedizione, al centro di evasione ordini e alle giurisdizioni di spedizione. Le regole di calcolo del gruppo di giurisdizioni di spedizione sono oggetti relazionali che associano le regole di calcolo della spedizione alle giurisdizioni, ai centri di evasione ordini e alle modalità di spedizione per determinare le regole di calcolo da utilizzare per ciascun articolo dell'ordine.

Se la regola di calcolo, o un altro oggetto a cui fanno riferimento le regole di calcolo del gruppo di giurisdizioni di spedizione viene cancellata, viene cancellata anche la regola di calcolo del gruppo di giurisdizioni di spedizione.



Per ulteriori informazioni sull'utilizzo delle regole di calcolo, consultare il manuale *IBM WebSphere Commerce Calculation Framework Guide*.

Giurisdizioni e gruppi di giurisdizioni

Le *giurisdizioni* sono aree o zone geografiche che rappresentano un paese o regione, una provincia o territorio oppure un codice postale a cui vengono vendute le merci. Le giurisdizioni sono raggruppate per formare *gruppi di giurisdizioni*.

WebSphere Commerce supporta due tipi di giurisdizioni: le giurisdizioni degli acquisti e le giurisdizioni delle tasse. Ciascuna di queste giurisdizioni fa parte di un gruppo corrispondente, ad esempio, le giurisdizioni di acquisti fanno parte del gruppo di giurisdizioni di spedizione e le giurisdizioni di tasse del gruppo di giurisdizioni di tasse.

I gruppi di giurisdizioni sono associati alle regole di calcolo. La regola di calcolo utilizza il gruppo di giurisdizioni come parte del calcolo per stabilire l'importo delle spese di spedizione.

Giurisdizioni e gruppi di giurisdizioni fanno parte di un'entità negozio. Se l'entità negozio viene cancellata, vengono cancellati anche le giurisdizioni ed i gruppi di giurisdizioni associati all'entità negozio.

Un indirizzo di spedizione può risolvere varie giurisdizioni di spedizione. Ad esempio, un indirizzo di spedizione di New York, negli Stati Uniti, verrà applicato alle seguenti giurisdizioni di spedizione: "New York, Stati Uniti", "Stati Uniti" e "Mondo". Quando un indirizzo di spedizione si applica a più giurisdizioni di spedizione, vengono applicate varie regole di calcolo della spedizione. In questi casi, verranno utilizzate le regole di calcolo del gruppo di giurisdizioni di spedizione associate per stabilire la regola o le regole da utilizzare.



Per informazioni più dettagliate sulla struttura delle risorse spedizione di WebSphere Commerce Server, fare riferimento ai modelli di oggetti e dati delle tasse nella guida in linea di WebSphere Commerce.

Creazione di risorse spedizione in WebSphere Commerce

Gli strumenti Servizi del negozio di WebSphere Commerce consentono di creare e modificare alcune ma non tutte le risorse spedizione (ad esempio le modalità di spedizione e le giurisdizioni) in un archivio negozio. Per ulteriori informazioni sulle risorse che è possibile modificare con gli strumenti Servizi del negozio, fare riferimento alla sezione *"Modifica delle risorse database del negozio"* della guida in linea di WebSphere Commerce.

Nota: Gli strumenti Servizi del negozio utilizzano file XML precostituiti sotto forma di un archivio negozio.

E' possibile creare le risorse spedizione in formato file XML che possono essere caricati nel database mediante il pacchetto Loader. Esistono due modi per creare le risorse spedizione:

- Modificare le risorse spedizione esistenti di uno dei negozi di esempio forniti con WebSphere Commerce o di un archivio negozio esistente
- Creare nuove risorse spedizione sotto forma di file XML

Per informazioni sulla modifica delle risorse spedizione di un archivio negozio esistente, fare riferimento alla guida in linea di WebSphere Commerce. Per informazioni sulla creazione di nuove risorse spedizione sotto forma di file XML, fare riferimento alla sezione “Creazione di risorse spedizione mediante un file XML”.

Creazione di risorse spedizione mediante un file XML

Creare le risorse spedizione in formato file XML che possono essere caricati nel database mediante il pacchetto Loader. Per ulteriori informazioni sul pacchetto Loader, consultare la Parte 7, “Pubblicazione del negozio” a pagina 213. Se il negozio da creare è multiculturale, è possibile creare file XML diversi per ogni locale supportata dal negozio. Il file specifico della locale deve contenere tutte le informazioni sulla descrizione, in modo che possa essere facilmente tradotto.

I negozi di esempio da cui sono ricavati molti esempi di queste attività, utilizzano un file shipping.xml per tutte le informazioni che non devono essere tradotte e un altro file shipping.xml per ogni locale supportata dal negozio, per le informazioni che devono essere tradotte. I file specifici della locale contengono tutte le informazioni sulla descrizione, in modo che possano essere facilmente tradotte.

Per creare le risorse spedizione per il negozio mediante un file XML, procedere come segue:

1. Rivedere il manuale *IBM WebSphere Commerce Calculation Framework Guide*. Il framework di calcolo di WebSphere Commerce calcola gli importi monetari (ad esempio quello relativo alla spedizione) associati al prodotto o servizio scelto per l'acquisto da un cliente.
2. Rivedere i file XML utilizzati per creare le risorse valuta nei negozi di esempio. Tutti i file per i negozi di esempio si trovano nel file dell'archivio negozio corrispondente. Ogni negozio di esempio presenta due o più file shipping.xml, che comprendono informazioni sulla spedizione. I file dell'archivio negozio si trovano nelle seguenti directory:

-  NT unità:\WebSphere\CommerceServer\samplstores
-  2000 unità:\Program Files\WebSphere\CommerceServer\samplstores
-  AIX /usr/WebSphere/CommerceServer/samplstores
-  Solaris /opt/WebSphere/CommerceServer/samplstores
-  Linux /opt/WebSphere/CommerceServer/samplstores
-  400 /qibm/proddata/WebCommerce/samplstores

Nota: La guida in linea di WebSphere Commerce contiene le informazioni su ciascuna delle risorse dati contenute nei negozi di esempio.

Per visualizzare i file shipping.xml nell'archivio negozio, decomprimerli con un programma ZIP. I file shipping.xml si trovano nella directory dei dati. Il file shipping.xml specifico della lingua si trova in una sottodirectory specifica della locale della directory dei dati.

3. Fare riferimento alle informazioni riportate in Appendice B, “Creazione di dati” a pagina 319.

4. Creare un file shipping.xml, copiando uno dei file shipping.xml presenti negli archivi negozio di esempio o creandone uno nuovo. Per ulteriori informazioni, fare riferimento al file DTD corrispondente a shipping.xml. I file DTD si trovano nelle seguenti directory:

-  unità:\WebSphere\CommerceServer\xml\sar
-  unità:\Program Files\WebSphere\CommerceServer\xml\sar
-  /usr/WebSphere/CommerceServer/xml/sar
-  /opt/WebSphere/CommercServer/xml/sar
-  /opt/WebSphere/CommercServer/xml/sar
-  /qibm/proddata/WebCommerce/xml/sar

5. Definire le giurisdizioni ed il gruppo di giurisdizioni a cui spedire merci e servizi. Tutte le giurisdizioni devono appartenere ad un gruppo di giurisdizioni.

a. Utilizzando il seguente esempio come guida, definire un gruppo di giurisdizioni nel file XML per la tabella JURSTGROUP:

```
<jurstgroup
  jurstgroup_id="@jurstgroup_id_1"
  description="Jurisdiction Group1 for Shipping"
  subclass="1"
  storeent_id="@storeent_id_1"
  code="World"/>
```

dove

- jurstgroup_id è una chiave univoca generata.
- description è una breve descrizione del gruppo di giurisdizioni, che può essere visualizzata in un'interfaccia utente che gestisce i gruppi di giurisdizioni.
- subclass è la sottoclasse del gruppo di giurisdizioni come riportato di seguito:
 - 1 = ShippingJurisdictionGroup
 - 2 = TaxJurisdictionGroup
- storeent_id è l'entità negozio associata a questo gruppo di giurisdizioni.
- code è un codice che, insieme alla relativa entità negozio e sottoclasse, identifica in maniera univoca questo gruppo di giurisdizioni.

b. Utilizzando il seguente esempio come guida, definire una giurisdizione nel file XML della tabella JURST.

```
< jurst
  jurst_id="@jurst_id_1"
  storeent_id="@storeent_id_1"
  code="World"
  subclass="1"/>
```

dove

- jurst_id è una chiave univoca generata.
- storeent_id è l'entità negozio associata a questo gruppo di giurisdizioni.
- code è un codice che, insieme alla relativa entità negozio e sottoclasse, identifica in maniera univoca questo gruppo di giurisdizioni.
- subclass è la sottoclasse della giurisdizione come riportato di seguito:

- 1 = ShippingJurisdiction
 - 2 = TaxJurisdiction
- c. Utilizzando il seguente esempio come guida, associare la giurisdizione creata al passo b con il gruppo di giurisdizioni definito al passo a, aggiungendo informazioni alla tabella JURSTGRPREL.

```
<jurstgprel
jurst_id="@jurst_id_1"
jurstgroup_id="@jurstgroup_id_1"
subclass="1"/>
dove
```

- jurst_id è la giurisdizione.
- jurstgroup_id è il gruppo di giurisdizioni
- subclass è la sottoclasse della giurisdizione e del gruppo di giurisdizioni, che deve corrispondere a:
 - 1 = ShippingJurisdiction[Group]
 - 2 = TaxJurisdiction[Group]

- d. Ripetere i passi da a a c per tutte le giurisdizioni e i gruppi di giurisdizioni supportati dal negozio.
6. Definire le modalità di spedizione utilizzate dal negozio.

- a. Utilizzando il seguente esempio come guida, definire una modalità di spedizione nel file XML per la tabella SHIPMODE:

```
<shipmode
shipmode_id="@shipmode_id_1"
field1
storeent_id="@storeent_id_1"
code="Ground 1 week"
carrier="XYZ Carrier"/>
dove:
```

- shipmode_id è una chiave univoca generata.
- field1 è un campo disponibile per la personalizzazione.
- storeent_id è l'entità negozio associata a questa modalità di spedizione.
- code è il codice assegnato al venditore, univoco per l'entità negozio.
- carrier è il nome o l'identificativo dello spedizioniere.

- b. Utilizzando il seguente esempio come guida, aggiungere informazioni sulla modalità di spedizione alla tabella SHPMODEDSC. Se il negozio da creare è multiculturale, è necessario includere queste informazioni in un file XML specifico delle locale:

```
<shpmodedsc
description="International mail"
field1="USD$5.00 per order plus USD$1.00 for each item"
field2="5 business days"
shipmode_id="@shipmode_id_1"
language_id="&en_US;" />
dove:
```

- description è una breve descrizione della modalità di spedizione che può essere visualizzata da un cliente per la selezione.
- field1 and field2 sono i campi disponibili per la personalizzazione.

- shipmode_id è una chiave univoca generata.
 - language_id è la lingua utilizzata.
- c. Ripetere i passi a e b per tutte le modalità di spedizione del negozio.
7. Definire i codici di calcolo utilizzati dal negozio.
- a. Utilizzando i seguenti esempi come guida, definire il codice di calcolo nel file XML per la tabella CALCODE.

```
< calcode
calcode_id="@calcode_id_1"
code="shipping Code 1- per/order"
calusage_id="-2"
storeent_id="@storeent_id_1"
groupby="0"
published="1"
sequence="+0.00E+000"
calmethod_id="-23"
calmethod_id_app="-24"
calmethod_id_qfy="-22"
flags="0" />
```

dove:

- calcode_id è una chiave univoca generata.
- code è una stringa di caratteri che identifica in modo univoco i dati codice di calcolo, utilizzo di calcolo e entità negozio.
- calusage_id indica il tipo di calcolo per cui viene utilizzato questo codice di calcolo. Ad esempio, il codice di calcolo può essere utilizzato per calcolare uno dei seguenti importi monetari:
 - Sconti (-1)
 - Spese di spedizione (-2)
 - Tasse di vendita (-3)
 - Tasse di spedizione (-4)
 - Coupon (-5)
- storeent_id è l'entità negozio associata a questo codice di calcolo.
- groupby sono indicatori che specificano nel metodo di combinazione dei codici di calcolo il modo in cui raggruppare gli articoli durante l'esecuzione dei calcoli. 0 = Nessun raggruppamento. Posiziona tutti gli articoli applicabili in un gruppo singolo. Per ulteriori informazioni, fare riferimento a *tabella CALCODE: dettagli* nella guida in linea di WebSphere Commerce.
- published specifica se il codice di calcolo è calcolato:
 - 0 = Non pubblicato (temporaneamente disabilitata)
 - 1 = Pubblicato
 - 2 = Contrassegnato per l'eliminazione (e non pubblicato)
- sequence è la sequenza con cui sono calcolati ed applicati i codici di calcolo, dal più basso al più alto. Se due codici di calcolo hanno lo stesso numero di sequenza, verranno calcolati per primi i codici di calcolo con il valore calcode_id più basso.
- calmethod_id è il metodo di calcolo del codice di calcolo che definisce come calcolare un importo monetario per il codice di calcolo.

calmethod_id="-23", il metodo di calcolo del codice di calcolo per la spedizione è l'unico metodo di calcolo per la spedizione fornito con WebSphere Commerce.

- calmethod_id_app è il metodo di applicazione del codice di calcolo che memorizza l'importo calcolato per gli articoli associati.
calmethod_id_app="-24", il metodo di applicazione del codice di calcolo per la spedizione è l'unico metodo di applicazione della spedizione fornito con WebSphere Commerce.
- calmethod_id_qfy è il metodo di qualifica del codice di calcolo che definisce gli articoli associati a questo codice di calcolo.
calmethod_id_qfy="-22", il metodo di qualifica del codice di calcolo per la spedizione è l'unico metodo di qualifica della spedizione fornito con WebSphere Commerce.
- flags specifica se è necessario richiamare il metodo di qualifica del codice di calcolo dello specifico codice di calcolo.
 - 0 = illimitato. Il metodo non sarà richiamato
 - 1 = limitato. Il metodo sarà richiamato

- b. Utilizzando il seguente esempio come guida, aggiungere le informazioni sulla descrizione del codice di calcolo nel file XML per la tabella CALCODEDSC. Se il negozio da creare è multiculturale, è necessario includere queste informazioni in un file XML specifico delle locale.

```
<calcodedsc  
  calcode_id="@calcode_id_3"  
  description="5.00USD per order"  
  language_id("&en_US"  
  longdescription="This shipping calculation code charges 5.00USD per order."  
>
```

dove

- calcode_id è il codice di calcolo a cui vengono applicate queste informazioni.
 - description è una descrizione breve del codice di calcolo.
 - language_id è la lingua a cui vengono applicate queste informazioni.
 - longdescription è la descrizione dettagliata del codice di calcolo.
- c. Ripetere i passi a e b per ciascun codice di calcolo utilizzato nel negozio.
8. Definire le regole di calcolo per il negozio.
- a. Utilizzando il seguente esempio come guida, impostare la regola di calcolo nel file XML per la tabella CALRULE:

```
<calrule  
  calrule_id="@calrule_id_1"  
  calcode_id="@calcode_id_1"  
  startdate="1900-01-01 00:00:00.000000"  
  enddate="2100-01-01 00:00:00.000000"  
  sequence="+1.0000000000000000E+000"  
  combination="2"  
  calmethod_id="-27"  
  calmethod_id_qfy="-26"  
  flags="1"  
  identifier="1" />dove
```

- calrule_id è un identificativo univoco generato.

- calcode_id è il codice di calcolo di cui fa parte questa regola di calcolo.
- startdate è l'ora in cui questa regola di calcolo diventa effettiva.
- enddate è l'ora in cui questa regola di calcolo non è più effettiva.
- sequence è l'ordine in cui verrà elaborata questa regola di calcolo. Le regole di calcolo per lo stesso codice di calcolo sono elaborate in sequenza dal valore più basso al più alto.
- combination specifica l'indicatore per l'elaborazione speciale che deve essere eseguita dall'implementazione del metodo di combinazione regole di calcolo predefinita. Per ulteriori informazioni, fare riferimento alla tabella CALRULE nella guida in linea di WebSphere Commerce.
- calmethod_id è il metodo di combinazione regole di calcolo mediante il quale viene calcolato l'importo per un insieme di articoli.
- calmethod_id_qfy è il metodo di qualifica regole di calcolo che determina l'insieme di articoli da inviare al metodo di calcolo regole di calcolo.
- flags vengono utilizzati dal metodo di combinazione regole di calcolo per determinare il modo in cui questa regola di calcolo può essere combinata con altre regole di calcolo. Per ulteriori informazioni, fare riferimento alla tabella CALRULE.
- identifier identifica questa regola di calcolo, insieme al relativo codice di calcolo.

Per ulteriori informazioni, fare riferimento alla tabella CALRULE nella guida in linea di WebSphere Commerce.

- b. Ripetere il passo a per ciascuna regola di calcolo utilizzata nel negozio. Ciascun codice di calcolo può avere diverse regole di calcolo. Ad esempio, calcode_id="@calcode_id_1" può essere associato a diversi calrule_id.
9. Definire la scala di calcolo per il negozio.

Una scala di calcolo è l'insieme di intervalli che si applicano al calcolo. Ad esempio, per i costi di spedizione si può disporre di un'insieme di intervalli di peso corrispondenti ciascuno a un determinato costo. Vale a dire che la spedizione di un prodotto che pesa tra 0 e 5 kg potrebbe costare 10 euro, mentre quella di un prodotto che pesa da 5 a 10 kg potrebbe costare 15 euro. Questi intervalli creano una scala.

- a. Utilizzando il seguente esempio come guida, impostare la scala di calcolo nel file XML per la tabella CALSCALE:

```
<calscale
calscale_id="@calscale_id_1"
code="Scale Code 1 per order USD"
storeent_id="@storeent_id_1"
calusage_id="-2"
setcurr="USD"
calmethod_id="-28"/>
dove
```

- calscale_id è un identificativo univoco generato.
- code è una stringa di caratteri che identifica in modo univoco questa scala di calcolo, dati un particolare utilizzo di calcolo e un'entità negozio.
- storeent_id è l'entità negozio di cui fa parte questa scala di calcolo.
- calusage_id indica il tipo di calcolo per cui viene utilizzata questa scala di calcolo. Ad esempio, la scala di calcolo può essere utilizzata per calcolare uno dei seguenti importi monetari:

- Sconti (-1)
- Spese di spedizione (-2)
- Tasse di vendita (-3)
- Tasse di spedizione (-4)
- Coupon (-5)
- setccurr se specificato, indica la valuta per i valori iniziali dell'intervallo degli oggetti di calcolo per questa regola di calcolo. Il metodo di ricerca scala di calcolo deve restituire un "numero di ricerca" in questa valuta.
- calmethod_id è il metodo di ricerca scala di calcolo che, dato un insieme di articoli dell'ordine, determina un valore di ricerca, un valore monetario di base, un moltiplicatore di risultato ed un insieme di misure matematiche che possono essere utilizzate dalla scala di calcolo per calcolare un importo. Per individuare il metodo di ricerca scala di calcolo da utilizzare, attenersi alla seguente procedura:
 - Fare riferimento alla tabella CALMETHOD nella guida in linea di WebSphere Commerce. Fare riferimento alla descrizione per la colonna SUBCLASS. Fare clic sul collegamento alla tabella CALMETHOD: dettagli. Questa tabella elenca i tipi di metodo di calcolo disponibili. Il metodo di ricerca scala di calcolo monetario è il numero 9.
 - Aprire il file bootstrap wcs.bootstrap_xx_XX.xml, dove xx_XX è il codice della locale. I file bootstrap si trovano nelle seguenti directory:
 -  NT unità:\WebSphere\CommerceServer\schema\xml
 -  2000 unità:\Program Files\WebSphere\CommerceServer\schema\xml
 -  AIX /usr/WebSphere/CommerceServer/schema/xml
 -  Solaris /opt/WebSphere/CommerceServer/schema/xml
 -  Linux /opt/WebSphere/CommerceServer/schema/xml
 -  400 /qibm/proddata/WebCommerce/schema/xml
 - Individuare la sezione contenente i metodi di calcolo disponibili (CALMETHOD).
 - Individuare i metodi di calcolo con il valore calusage_ID per le tasse (-3 per le tasse di vendita e -4 per le tasse di acquisto).
 - Individuare i metodi di calcolo che hanno una sottoclasse di 7 (ne esistono molti). Selezionare il metodo di calcolo che risponde alle proprie necessità.

Per ulteriori informazioni, fare riferimento alla tabella CALSCALE nella guida in linea di WebSphere Commerce.

- b. Ripetere il passo a per ciascuna scala di calcolo utilizzata nel negozio. Ad esempio, per la spedizione, InFashion crea un costo per la scala degli ordini e un costo per la scala degli articoli.
10. Definire gli intervalli di calcolo per le scale di calcolo.
 - a. Utilizzando il seguente esempio come guida, impostare l'intervallo di calcolo nel file XML per la tabella CALRANGE.

```
<calrange
calrange_id="@calrange_id_1"
calscale_id="@calscale_id_1"
calmethod_id="-33"
```

```
rangestart="0.00000"  
cumulative="0"/>
```

dove

- calrange_id è un identificativo univoco generato.
- calscale_id è la scala di calcolo di cui fa parte questo intervallo di calcolo.
- calmethod_id è il metodo di intervallo del calcolo che restituisce un importo monetario come risultato della ricerca dell'intervallo del calcolo. Ad esempio, FixedAmountCalculationRangeCmd, PerUnitAmountCalculationRangeCmd oppure PercentageCalculationRangeCmd. Per stabilire il metodo di intervallo del calcolo, attenersi alla seguente procedura:
 - Fare riferimento alla tabella CALMETHOD nella guida in linea di WebSphere Commerce. Fare riferimento alla descrizione per la colonna SUBCLASS. Fare clic sul collegamento alla tabella CALMETHOD: dettagli. Questa tabella elenca i tipi di metodo di calcolo disponibili. Il metodo di intervallo del calcolo è il numero 10.
 - Aprire il file bootstrap wcs.bootstrap_xx_XX.xml, dove xx_XX è il codice della locale. I file bootstrap si trovano nelle seguenti directory:
 -  NT unità:\WebSphere\CommerceServer\schema\xml
 -  2000 unità:\Program Files\WebSphere\CommerceServer\schema\xml
 -  AIX /usr/WebSphere/CommerceServer/schema/xml
 -  Solaris /opt/WebSphere/CommerceServer/schema/xml
 -  Linux /opt/WebSphere/CommerceServer/schema/xml
 -  400 /qibm/proddata/WebCommerce/schema/xml
 - Individuare la sezione contenente i metodi di calcolo disponibili (CALMETHOD).
 - Individuare i metodi di calcolo con il valore calusage_ID per le tasse (-3 per le tasse di vendita e -4 per le tasse di acquisto).
 - Individuare i metodi di calcolo che hanno una sottoclasse di 9 (ne esistono molti). Selezionare il metodo di calcolo che risponde alle proprie necessità.
- cumulative sono i valori validi:
 - 0 = viene utilizzato solo l'intervallo di calcolo corrispondente con il valore RANGESTART più elevato.
 - 1 = vengono utilizzati tutti gli intervalli di calcolo corrispondenti. Gli importi calcolati vengono sommati in modo da ottenere il risultato finale.

Per ulteriori informazioni, fare riferimento alla tabella CALRANGE nella guida in linea di WebSphere Commerce.

- b. Ripetere il passo a per ciascun intervallo di calcolo associato alla scala di calcolo utilizzata nel negozio.
11. Definire i valori di ricerca del calcolo per le scale di calcolo. I valori di ricerca del calcolo sono i valori associati alla scala di calcolo. Ad esempio, una scala di calcolo include gli intervalli di peso e i relativi prezzi per la spedizione di seguito riportati:

- 0 - 5 kg al costo di 10 euro
- 5 - 10 kg al costo di 15 euro

I valori di ricerca sono 10 e 15 euro

- a. Utilizzando i seguenti esempi come guida, impostare la ricerca di calcolo nel file XML per la tabella CALRLOOKUP. Se il negozio da creare è multiculturale, è necessario includere queste informazioni in un file XML specifico delle locale, vale a dire in un file per ciascuna locale supportata dal negozio. Ad esempio, se il negozio effettua spedizioni a clienti negli Stati Uniti e in Giappone, è necessario aggiungere i valori di ricerca del dollaro americano in un file XML ed i valori di ricerca dello Yen in un altro file XML.

```
<calrlookup
calrlookup_id="@calrlookup_id_1"
setccurr="USD"
calrange_id="@calrange_id_1"
value="5.00"/>
```

dove

- calrlookup_id è un identificativo univoco generato.
- calrange_id è l'intervallo di calcolo di cui fa parte il risultato della ricerca dell'intervallo di calcolo.
- value è il valore del risultato della ricerca dell'intervallo di calcolo, utilizzato dal metodo dell'intervallo di calcolo per stabilire un importo.

Per ulteriori informazioni, fare riferimento alla tabella CALRLOOKUP nella guida in linea di WebSphere Commerce.

- b. Ripetere il passo a per ciascun valore di ricerca associato alla scala di calcolo utilizzata nel negozio.
12. Associare la regola di calcolo e la scala di calcolo.

- a. Utilizzando i seguenti esempi come guida, associare la scala di calcolo alla regola di calcolo nel file XML per la tabella CRULESCALE.

```
< crulescale
calrule_id="@calrule_id_1"
calscale_id="@calscale_id_1" />
```

dove

- calrule_id è la regola di calcolo.
- calscale_id è la scala di calcolo.

- b. Ripetere il passo a per ciascuna associazione di scala di calcolo e regola.



Per ulteriori informazioni sull'uso di @ e &, fare riferimento all'Appendice B, "Creazione di dati" a pagina 319.

Creazione di risorse evasione ordini per la spedizione

Per fare in modo che le risorse spedizione funzionino correttamente nel negozio, è necessario associare i gruppi di giurisdizioni di spedizione alle regole di calcolo e i centri di evasione ordine alle modalità di spedizione utilizzate nel negozio.

Per poter associare le risorse spedizione ad un centro di evasione ordini, è necessario creare le risorse evasione ordini. Per ulteriori informazioni sulla

creazione delle risorse evasione ordini, fare riferimento alla sezione “Creazione di risorse evasione ordini in WebSphere Commerce” a pagina 112.

Una volta create le risorse evasione ordini, associarvi le risorse spedizione, aggiungendo informazioni alle tabelle SHPJCRULE e SHPARRANGE. Attenersi alla procedura di seguito riportata:

1. Rivedere il manuale *IBM WebSphere Commerce Calculation Framework Guide*. Il framework di calcolo di WebSphere Commerce calcola gli importi monetari (ad esempio quello relativo alla spedizione) associati al prodotto o servizio scelto per l’acquisto da un cliente.
2. Rivedere i file XML utilizzati per creare le risorse evasione ordini relative alla spedizione per i negozi di esempio. Tutti i file per i negozi di esempio si trovano nel file dell’archivio negozio corrispondente. Ogni negozio di esempio presenta un file `shipfulfill.xml`, che comprende informazioni sull’evasione ordini relativa alla spedizione. Per visualizzare il file `shipfulfill.xml` nell’archivio negozio, decomprimere il file con un programma ZIP. Il file `shipfulfill.xml` si trova nella directory dei dati.

I file dell’archivio negozio si trovano nelle seguenti directory:

-  unità:\WebSphere\CommerceServer\samplstores
-  unità:\Program Files\WebSphere\CommerceServer\samplstores
-  /usr/WebSphere/CommerceServer/samplstores
-  /opt/WebSphere/CommerceServer/samplstores
-  /opt/WebSphere/CommerceServer/samplstores
-  /qibm/proddata/WebCommerce/samplstores

Nota: La guida in linea di WebSphere Commerce contiene le informazioni su ciascuna delle risorse dati contenute nei negozi di esempio.

3. Fare riferimento alle informazioni riportate in Appendice B, “Creazione di dati” a pagina 319.
4. Creare un file `shipfulfill.xml`, copiando uno dei file `shipfulfill.xml` presenti negli archivi negozio di esempio o creandone uno nuovo. Per ulteriori informazioni, fare riferimento al file DTD corrispondente a `shipfulfill.xml`. I file DTD si trovano nelle seguenti directory:

-  unità:\WebSphere\CommerceServer\xml\sar
-  unità:\Program Files\WebSphere\CommerceServer\xml\sar
-  /usr/WebSphere/CommerceServer/xml/sar
-  /opt/WebSphere/CommerceServer/xml/sar
-  /opt/WebSphere/CommerceServer/xml/sar
-  /qibm/proddata/WebCommerce/xml/sar

5. Associare le regole di calcolo ad un gruppo di giurisdizioni di spedizione aggiungendo informazioni alla tabella SHPJCRULE. Utilizzare il seguente esempio come guida: Se il negozio da creare è multiculturale, creare anche un file XML per ogni locale supportata dal negozio.

```
<shpjcrule  
calrule_id="@calrule_id_1"  
ffmcenter_id="@ffmcenter_id_1"
```

```
jurstgroup_id="@jurstgroup_id_1"
precedence="0"
shipmode_id="@shipmode_id_1"
shpjcrule_id="@shpjcrule_id_1"
```

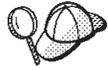
dove

- calrule_id è la regola di calcolo utilizzata.
 - ffmcenter_id è il centro di evasione ordini. Se è NULL, questa associazione si applica a tutti i centri di evasione ordini.
 - jurstgroup_id è il gruppo di giurisdizioni di spedizione. Se è NULL, questa associazione si applica a tutti i gruppi di giurisdizioni di spedizione.
 - precedence si applica quando un indirizzo di spedizione è compreso in più gruppi di giurisdizioni di spedizione specificati per lo stesso centro di evasione ordini e la stessa modalità di spedizione. Viene applicata solo la regola di calcolo con il valore SHPJCRULE.PRECEDENCE più alto.
 - shipmode_id è la modalità di spedizione.
 - shpjcrule_id è un identificativo univoco generato.
6. Ripetere il passo 3 per ciascuna associazione tra gruppi di giurisdizioni, centri di evasione ordini e regole.
 7. Associare una modalità di spedizione e un centro di evasione ordini al negozio, aggiungendo informazioni alla tabella SHPARRANGE. Utilizzare il seguente esempio come guida:

```
<shparrange
shparrange_id="@shparrange_id_2"
store_id="@storent_id_1"
ffmcenter_id="@ffmcenter_id_1"
shipmode_id="@shipmode_id_2"
startdate="1970-06-22 23:00:00.000000"
enddate="2008-06-22 23:00:00.000000"
precedence="0"
flags="0"
/>
```

dove

- shparrange_id è un identificativo univoco generato.
 - store_id è il negozio.
 - ffmcenter_id è il centro di evasione ordini.
 - shipmode_id è la modalità di spedizione. NULL indica la disposizione di spedizione che può essere utilizzata indipendentemente dalla modalità di spedizione.
 - startdate è l'ora in cui questa disposizione di spedizione diventa effettiva.
 - enddate è l'ora in cui questa disposizione di spedizione non è più effettiva.
 - precedence è il valore di precedenza relativo alle disposizioni di spedizione. Quando più disposizioni di spedizione (per lo stesso negozio e modalità di spedizione) sono valide in un determinato momento, viene utilizzata quella con il valore PRECEDENCE maggiore.
 - flags contiene i seguenti indicatori:
 - 1 = restricted - Questa disposizione di spedizione si applica solo agli articoli dell'ordine i cui indirizzi di spedizione corrispondono a uno dei gruppi di giurisdizione di spedizione associati (mediante la tabella SHPARJURGP) con la disposizione.
8. Ripetere il passo 5 per tutte le modalità di spedizione utilizzate nel negozio.



Creazione di risorse spedizione catalogo del negozio

Per associare le modalità di spedizione al negozio, è necessario associare un codice di calcolo alle voci di catalogo del negozio per ciascun contratto del negozio.

Per poter creare le risorse spedizione catalogo del negozio, è necessario creare le risorse del negozio e del catalogo. Per ulteriori informazioni sulla creazione delle risorse del negozio, fare riferimento alla sezione "Creazione di risorse dati del negozio in un file XML" a pagina 45. Per ulteriori informazioni sulla creazione delle risorse catalogo, consultare la sezione "Visualizzazione delle risorse catalogo del negozio" a pagina 77.

Per creare le risorse spedizione catalogo del negozio, procedere come segue:

1. Rivedere il manuale *IBM WebSphere Commerce Calculation Framework Guide*. Il framework di calcolo di WebSphere Commerce calcola gli importi monetari (ad esempio quello relativo alla spedizione) associati al prodotto o servizio scelto per l'acquisto da un cliente.
2. Rivedere i file XML utilizzati per creare le risorse evasione ordini relative alla spedizione per i negozi di esempio. Tutti i file per i negozi di esempio si trovano nel file dell'archivio negozio corrispondente.

I file dell'archivio negozio si trovano nelle seguenti directory:

-  NT unità:\WebSphere\CommerceServer\samplestores
-  2000 unità:\Program Files\WebSphere\CommerceServer\samplestores
-  AIX /usr/WebSphere/CommerceServer/samplestores
-  Solaris /opt/WebSphere/CommerceServer/samplestores
-  Linux /opt/WebSphere/CommerceServer/samplestores
-  400 /qibm/proddata/WebCommerce/samplestores

Nota: La guida in linea di WebSphere Commerce contiene le informazioni su ciascuna delle risorse dati contenute nei negozi di esempio.

Ogni negozio di esempio presenta un file `store-catalog-shipping.xml`, che comprende informazioni sull'evasione ordini relativa alla spedizione. Per visualizzare il file `store-catalog-shipping.xml` nell'archivio negozio, decomprimere il file con un programma ZIP. Il file `store-catalog-shipping.xml` si trova nella directory dei dati.

3. Fare riferimento alle informazioni riportate in Appendice B, "Creazione di dati" a pagina 319.
4. Creare un file `store-catalog-shipping.xml`, copiando uno dei file `store-catalog-shipping.xml` presenti negli archivi negozio di esempio o creandone uno nuovo. Per ulteriori informazioni, fare riferimento al file DTD corrispondente a `store-catalog-shipping.xml`. I file DTD si trovano nelle seguenti directory:

-  NT unità:\WebSphere\CommerceServer\xml\sar
-  2000 unità:\Program Files\WebSphere\CommerceServer\xml\sar

-  /usr/WebSphere/CommerceServer/xml/sar
 -  /opt/WebSphere/CommerceServer/xml/sar
 -  /opt/WebSphere/CommerceServer/xml/sar
 -  /qibm/proddata/WebCommerce/xml/sar
5. Creare la relazione spedizione catalogo del negozio aggiungendo informazioni alla tabella CATENCALCD. Utilizzare il seguente esempio come guida:

```
<catencalcd
  calcode_id="@calcode_id_1"
  catencalcd_id="@catencalcd_id_1"
  store_id="@storeent_id_1"
/>
```

dove

- calcode_id è il codice di calcolo.
- catencalcd_id è un identificativo univoco generato.
- store_id è il negozio.



Per ulteriori informazioni sull'uso di @ e &, fare riferimento all'Appendice B, "Creazione di dati" a pagina 319.

Creazione di una modalità di spedizione predefinita

Per impostare una modalità di spedizione predefinita per il negozio, è necessario aggiungere informazioni alla tabella STOREDEF. Per aggiungere informazioni alla tabella STOREDEF, attenersi alla seguente procedura:

1. Rivedere i file XML utilizzati per creare le risorse predefinite del negozio per i negozi di esempio. Tutti i file per i negozi di esempio si trovano nel file dell'archivio negozio corrispondente.

I file dell'archivio negozio si trovano nelle seguenti directory:

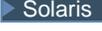
-  unità:\WebSphere\CommerceServer\samplestores
-  unità:\Program Files\WebSphere\CommerceServer\samplestores
-  /usr/WebSphere/CommerceServer/samplestores
-  /opt/WebSphere/CommerceServer/samplestores
-  /opt/WebSphere/CommerceServer/samplestores
-  /qibm/proddata/WebCommerce/samplestores

Nota: La guida in linea di WebSphere Commerce contiene le informazioni su ciascuna delle risorse dati contenute nei negozi di esempio.

Ogni negozio di esempio presenta un file store-defaults.xml, che comprende informazioni sulla spedizione predefinita. Per visualizzare il file store-defaults.xml nell'archivio negozio, decomprimere il file con un programma ZIP. Il file store-defaults.xml si trova nella directory dei dati.

2. Fare riferimento alle informazioni riportate in Appendice B, "Creazione di dati" a pagina 319.
3. Creare un file store-defaults.xml, copiando uno dei file store-defaults.xml presenti negli archivi negozio di esempio o creandone uno nuovo. Per ulteriori

informazioni, fare riferimento al file DTD corrispondente a store-defaults.xml. I file DTD si trovano nelle seguenti directory:

-  unità:\WebSphere\CommerceServer\xml\sar
 -  unità:\Program Files\WebSphere\CommerceServer\xml\sar
 -  /usr/WebSphere/CommerceServer/xml/sar
 -  /opt/WebSphere/CommerceServer/xml/sar
 -  /opt/WebSphere/CommerceServer/xml/sar
 -  /qibm/proddata/WebCommerce/xml/sar
4. Utilizzando il seguente esempio come guida, nel file XML specificare la modalità di spedizione predefinita per il negozio aggiungendo informazioni alla tabella STOREDEF:

```
<storedef
  store_id="@storeent_id_1"
  shipmode_id="@shipmode_id_1"
/>
```

dove

- store_id è il negozio.
- shipmode_id è la modalità di spedizione predefinita per il negozio.



Per ulteriori informazioni sull'uso di @ e &, fare riferimento all'Appendice B, "Creazione di dati" a pagina 319.

Capitolo 19. Risorse tasse

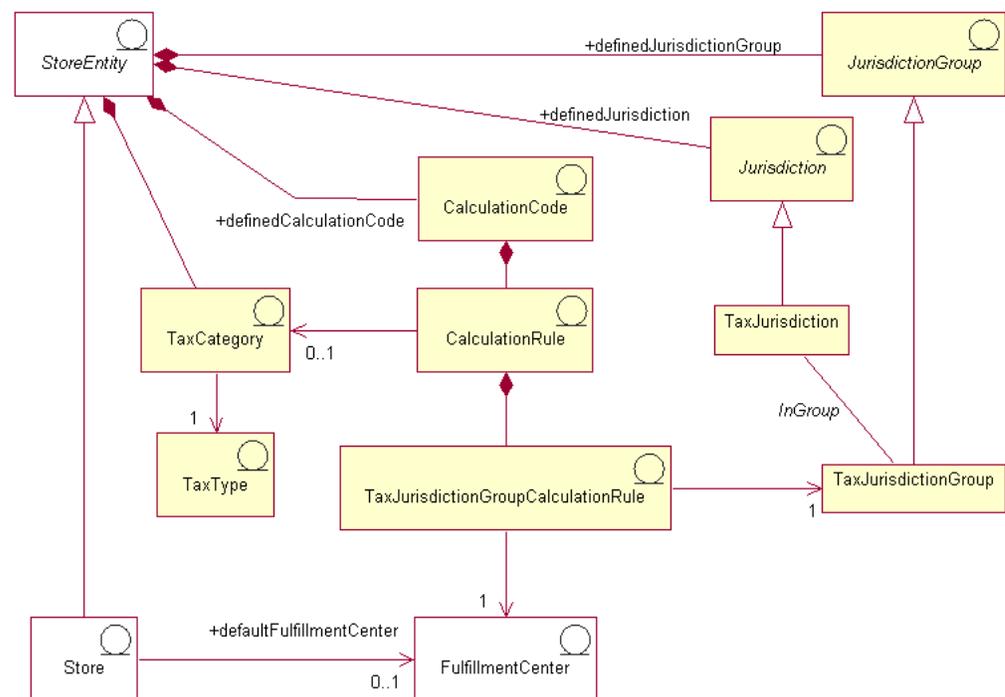
Per calcolare e raccogliere le tasse sulle merci e i servizi forniti dal negozio, un negozio creato con WebSphere Commerce deve includere quanto di seguito riportato:

- Categorie di tassa
- Codici di calcolo
- Giurisdizioni e gruppi di giurisdizioni

La combinazione delle categorie di tasse, dei codici di calcolo e delle giurisdizioni e gruppi di giurisdizioni crea le tasse del negozio.

Risorse tasse in WebSphere Commerce

Il grafico di seguito riportato illustra la struttura della tassazione in WebSphere Commerce Server.



Questo grafico, insieme a tutti gli altri grafici della sezione relativa ai dati del negozio fa parte del modello di informazioni di WebSphere Commerce Server. Per ulteriori informazioni sul modello di informazioni, fare riferimento alla sezione "Modello di informazioni dati del negozio" a pagina 27. Per ulteriori informazioni sulle convenzioni utilizzate in questo grafico, fare riferimento all'Appendice A, "Legenda UML" a pagina 317.

Categorie di tasse

Le *categorie di tasse* corrispondono ai diversi tipi di tasse che un negozio può richiedere, come tasse federali, statali, provinciali e comunali.

Una categoria di tasse è parte di un'entità negozio, anche se un'entità negozio può avere diverse categorie di tasse. Se l'entità negozio viene cancellata, vengono cancellate anche le categorie di tasse associate all'entità negozio.

Tipo di tasse

Un negozio raccoglie in genere due tipi di tasse: le tasse di vendita o tasse sull'uso e le tasse di spedizione. Ciascuna categoria di tasse presenta un *tipo di tassa*. Nonostante le categorie di tasse possano presentare un unico tipo di tassa (ad esempio la categoria di tasse federali presenta la tassa di tipo vendita), più categorie di tasse diverse possono appartenere allo stesso tipo di tassa (ad esempio, la tassa di vendita viene applicata alle categorie federale, provinciale e comunale).

Codice di calcolo

I *codici di calcolo* vengono utilizzati per calcolare l'addebito delle tasse, vale a dire che un codice di calcolo indica il modo in cui le tasse vengono calcolate per gli articoli dell'ordine. Per calcolare le tasse per un articolo dell'ordine, è necessario assegnare i codici di calcolo delle tasse di spedizione e di vendita a una voce di catalogo o ad un gruppo di voci di catalogo. Un solo codice di calcolo di ciascun tipo di tassa può essere applicato ad una particolare voce di catalogo o gruppo di voci di catalogo. In genere, la tassa di vendita e d'uso è calcolata sul prezzo netto e la tassa di spedizione sulle spese di spedizione.

Un codice di calcolo è parte di un'entità negozio. Un codice di calcolo può essere associato ad un'unica entità negozio, ma un negozio può avere diversi codici di calcolo. Se l'entità negozio viene cancellata, vengono cancellati anche i codici di calcolo associati all'entità negozio.



Per ulteriori informazioni sull'utilizzo dei codici di calcolo, consultare il manuale *IBM WebSphere Commerce Calculation Framework Guide*.

Regole di calcolo

Ogni codice di calcolo presenta almeno una *regola di calcolo*, che definisce i calcoli per ciascuna categoria di tasse e specifica le condizioni in cui verranno eseguiti i calcoli. Ciascuna regola di calcolo delle tasse è associata a una categoria di tasse, a un gruppo di giurisdizioni e ad un centro di evasione ordini, che definiscono le condizioni in cui viene utilizzata la regola di calcolo. Ad esempio, una regola diversa può essere selezionata per calcolare l'importo di una particolare categoria di tasse, in base all'indirizzo di spedizione e al centro di evasione ordini specificato nell'ordine.

Ogni regola di calcolo appartiene ad un esatto codice di calcolo.

Un determinato codice di calcolo delle tasse può presentare diverse regole di calcolo, una per ciascuna combinazione di categoria di tasse, gruppo di giurisdizione tasse e centro di evasione ordini associata al negozio. Ciascuna regola di calcolo delle tasse di vendita e di spedizione può essere associata a più `TaxJurisdictionGroupCalculationRules` (`TaxRules`). Ad esempio, nel grafico di seguito riportato, la regola di calcolo 10001 è applicabile ai gruppi di giurisdizioni 1234 e 1235.

TAXJCRULE_ID	CALRULE_ID	FFMCENTER_ID	JURSTGROUP_ID	PRECEDENCE
10001	10001	NULL	1234	0
10002	10001	NULL	1235	0

Ciascuna TaxRule definisce le condizioni in cui la regola di calcolo deve essere applicata. Ad esempio, è possibile definire una regola di calcolo per ciascun gruppo di giurisdizioni a cui il negozio invia merce. Nell'esempio di seguito riportato, la regola di calcolo 10001 è applicabile al gruppo di giurisdizioni 1234 e 1235.

Nell'esempio di seguito riportato, il codice di calcolo delle tasse utilizza la regola di calcolo A per la categoria di tasse di vendita provinciale, quando la giurisdizione di tasse è Alberta e la regola C quando la giurisdizione di tasse è British Columbia.

Giurisdizione tasse	Tasse di vendita federale	Tasse di vendita provinciale
Alberta, Canada	Regola di calcolo B, con risultato Y%	Regola di calcolo B, con risultato X%
British Columbia, Canada	Regola di calcolo B, con risultato Y%	Regola di calcolo C, con risultato Z%

Quando un indirizzo di spedizione corrisponde a più di un gruppo di giurisdizioni, viene utilizzata la regola di calcolo con il più alto valore di colonna TAXJCRULE.PRECEDENCE associato.

L'associazione di TaxJurisdictionGroupCalculationRules (TaxRule) con una regola di calcolo determina il momento in cui è applicabile la regola di calcolo. Una regola di calcolo per la tassa di vendita o di spedizione è applicabile quando vengono soddisfatte tutte le condizioni date da TaxRules. Nell'esempio di seguito riportato, la regola di calcolo 10001 è applicabile per la spedizione al gruppo di giurisdizioni 1001 o per la spedizione dal centro di evasione ordini 1001.

CALRULE_ID	FFMCENTER_ID	JURSTGROUP_ID
10001	NULL	1001
10001	1001	1001

Ciascuna TaxJurisdictionGroupCalculationRule è associata con al massimo un gruppo di giurisdizioni. Le regole di calcolo non sono direttamente associate ai gruppi di giurisdizioni.



Per ulteriori informazioni sull'utilizzo delle regole di calcolo, consultare il manuale *IBM WebSphere Commerce Calculation Framework Guide*.

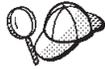
Giurisdizioni e gruppi di giurisdizioni

Le *giurisdizioni* sono aree o zone geografiche che rappresentano un paese o regione, una provincia o territorio oppure un codice postale a cui vengono vendute le merci. Le giurisdizioni sono raggruppate per formare *gruppi di giurisdizioni*.

WebSphere Commerce supporta due tipi di giurisdizioni: le giurisdizioni di spedizione e le giurisdizioni di tasse. Ciascuna di queste giurisdizioni fa parte di un gruppo corrispondente, ad esempio, le giurisdizioni di spedizione fanno parte del gruppo di giurisdizioni di spedizione e le giurisdizioni di tasse del gruppo di giurisdizioni di tasse.

Le giurisdizioni e i gruppi di giurisdizioni determinano le regole di calcolo utilizzate per calcolare l'importo delle tasse.

Giurisdizioni e gruppi di giurisdizioni fanno parte di un'entità negozio. Ciascuna giurisdizione e gruppo di giurisdizione è parte di un'entità negozio, anche se un'entità negozio può avere diverse giurisdizioni o gruppi di giurisdizioni. Se l'entità negozio viene cancellata, vengono cancellati anche le giurisdizioni ed i gruppi di giurisdizioni associati all'entità negozio.



Per informazioni più dettagliate sulla struttura delle risorse tasse di WebSphere Commerce Server, fare riferimento ai modelli di oggetti e dati delle tasse nella guida in linea di WebSphere Commerce.

Creazione di risorse tasse in WebSphere Commerce

Gli strumenti Servizi del negozio di WebSphere Commerce consentono di creare e modificare alcune ma non tutte le risorse tasse (ad esempio categorie di tasse e giurisdizioni) in un archivio negozio. Per ulteriori informazioni sulle risorse che è possibile modificare con gli strumenti Servizi del negozio, fare riferimento alla sezione *"Modifica delle risorse database del negozio"* della guida in linea di WebSphere Commerce.

Nota: Gli strumenti Servizi del negozio utilizzano file XML precostituiti sotto forma di un archivio negozio.

E' possibile creare le risorse tasse in formato file XML che possono essere caricati nel database mediante il pacchetto Loader. Esistono due modi per creare le risorse tasse:

- Modificare le risorse tasse esistenti di uno dei negozi di esempio forniti con WebSphere Commerce o di un archivio negozio esistente
- Creare nuove risorse tasse sotto forma di un file XML, che può essere pubblicato come parte di un archivio negozio o caricato mediante il Pacchetto Loader.

Per informazioni sulla modifica delle risorse tasse di un archivio negozio esistente, fare riferimento alla guida in linea di WebSphere Commerce. Per informazioni sulla creazione di nuove risorse tasse sotto forma di file XML, fare riferimento alla sezione *"Creazione di risorse tasse mediante un file XML"*.

Creazione di risorse tasse mediante un file XML

Creare le risorse tasse in formato file XML che possono essere caricati nel database mediante il pacchetto Loader. Per ulteriori informazioni sul pacchetto Loader, consultare la Parte 7, *"Pubblicazione del negozio"* a pagina 213. Se il negozio da creare è multiculturale, è possibile creare file XML diversi per ogni locale supportata dal negozio. Il file specifico delle locale deve contenere tutte le informazioni sulla descrizione, in modo che possa essere facilmente tradotto.

I negozi di esempio, da cui sono ricavati molti esempi di queste attività, utilizzano un file `tax.xml` per tutte le informazioni che non devono essere tradotte e un altro file `tax.xml` per ciascuna locale supportata dal negozio, per le informazioni che devono essere tradotte. I file specifici delle locale contengono tutte le informazioni sulla descrizione.

Per creare le risorse tasse per il negozio mediante un file XML, procedere come segue:

1. Rivedere il manuale *IBM WebSphere Commerce Calculation Framework Guide*. Il framework di calcolo di WebSphere Commerce calcola gli importi monetari (ad esempio quello relativo alle tasse) associati al prodotto o servizio scelto per l'acquisto da un cliente.

2. Rivedere i file XML utilizzati per creare le risorse tasse per i negozi di esempio. Tutti i file per i negozi di esempio si trovano nel file dell'archivio negozio corrispondente.

I file dell'archivio negozio si trovano nelle seguenti directory:

-  NT unità:\WebSphere\CommerceServer\samplstores
-  2000 unità:\Program Files\WebSphere\CommerceServer\samplstores
-  AIX /usr/WebSphere/CommerceServer/samplstores
-  Solaris /opt/WebSphere/CommerceServer/samplstores
-  Linux /opt/WebSphere/CommerceServer/samplstores
-  400 /qibm/proddata/WebCommerce/samplstores

Nota: La guida in linea di WebSphere Commerce contiene le informazioni su ciascuna delle risorse dati contenute nei negozi di esempio.

Ciascun negozio di esempio presenta due file `tax.xml`, contenenti le informazioni sulle tasse. Per visualizzare i file `tax.xml` nell'archivio negozio, decomprimere il file con un programma ZIP. I file `tax.xml` si trovano nella directory dei dati. Il file `tax.xml` specifico della lingua si trova in una sottodirectory specifica delle locale della directory dei dati.

3. Fare riferimento alle informazioni riportate in Appendice B, "Creazione di dati" a pagina 319.
4. Creare un file `tax.xml`, copiando uno dei file `tax.xml` presenti negli archivi negozio di esempio o creandone uno nuovo. Per ulteriori informazioni, fare riferimento al file DTD corrispondente a `tax.xml`. I file DTD si trovano nelle seguenti directory:

-  NT unità:\WebSphere\CommerceServer\xml\sar
-  2000 unità:\Program Files\WebSphere\CommerceServer\xml\sar
-  AIX /usr/WebSphere/CommerceServer/xml/sar
-  Solaris /opt/WebSphere/CommerceServer/xml/sar
-  Linux /opt/WebSphere/CommerceServer/xml/sar
-  400 /qibm/proddata/WebCommerce/xml/sar

5. Definire le giurisdizioni e i gruppi di giurisdizioni a cui spedire merci e servizi. Assegnare le giurisdizioni tasse ai gruppi di giurisdizioni tasse in base alle regole di calcolo della categoria di tasse applicabili.

- a. Utilizzando il seguente esempio come guida, definire un gruppo di giurisdizioni nel file XML per la tabella JURSTGROUP:

```
<jurstgroup
jurstgroup_id="@jurstgroup_id_2"
description="Tax Jurstiction Group 1"
subclass="2"
storeent_id="@storeent_id_1"
code="World"/>
```

dove

- `jurstgroup_id` è una chiave univoca generata.

- description è una breve descrizione del gruppo di giurisdizioni, che può essere visualizzata in un'interfaccia utente che gestisce i gruppi di giurisdizioni.
 - subclass è la sottoclasse del gruppo di giurisdizioni come riportato di seguito:
 - 1 = ShippingJurisdictionGroup
 - 2 = TaxJurisdictionGroup
 - storeent_id è l'entità negozio associata a questo gruppo di giurisdizioni.
 - code è un codice che, insieme alla relativa entità negozio e sottoclasse, identifica in maniera univoca questo gruppo di giurisdizioni.
- b. Utilizzando il seguente esempio come guida, definire una giurisdizione nel file XML della tabella JURST.

```
<jurst
  jurst_id="@jurst_id_2"
  storeent_id="@storeent_id_1"
  code="World"
  subclass="2"/>
```

dove

- jurst_id è una chiave univoca generata.
 - storeent_id è l'entità negozio associata a questo gruppo di giurisdizioni.
 - code è un codice che, insieme alla relativa entità negozio e sottoclasse, identifica in maniera univoca questo gruppo di giurisdizioni.
 - subclass è la sottoclasse della giurisdizione come riportato di seguito:
 - 1 = ShippingJurisdiction
 - 2 = TaxJurisdiction
- c. Utilizzando il seguente esempio come guida, associare la giurisdizione creata al passo b con il gruppo di giurisdizioni definito al passo a, aggiungendo informazioni alla tabella JURSTGRPTEL.

```
<jurstgrpTEL
  jurst_id="@jurst_id_2"
  jurstgroup_id="@jurstgroup_id_1"
  subclass="2"/>
```

dove

- jurst_id è la giurisdizione.
 - jurstgroup_id è il gruppo di giurisdizioni.
 - subclass è la sottoclasse della giurisdizione e del gruppo di giurisdizioni, che deve corrispondere a:
 - 1 = ShippingJurisdiction[Group]
 - 2 = TaxJurisdiction[Group]
- d. Ripetere i passi da a a c per tutte le giurisdizioni e i gruppi di giurisdizioni supportati dal negozio.

6. Definire le categorie di tasse utilizzate dal negozio.

- a. Utilizzando il seguente esempio come guida, definire una categoria di tasse nel file XML per la tabella TAXCGRY:

```
<taxcgrY
  taxcgrY_id="@taxcgrY_id_1"
  taxtype_id="-3"
```

```

storeent_id="@storeent_id_1"
name="Sales Tax"
displayseq="0"
displayusage="0"/>

```

dove:

- taxcgr_id è una chiave univoca generata.
- taxtype_id="-3" è il tipo di tassa per questa categoria di tasse. WebSphere Commerce supporta due tipi di tasse:
 - tasse di vendita o di utilizzo (-3)
 - tasse di spedizione (-4)
- storeent_id è l'entità negozio associata a questa categoria di tasse.
- name è il nome della categoria di tasse. Insieme all'entità negozio, il nome identifica in modo univoco questa categoria di tasse.
- displayseq specifica la sequenza, dal valore più basso a quello più alto, dell'importo delle tasse quando visualizzato, ad esempio, in un ordine.
- displayusage specifica che questa categoria di tasse, in relazione al valore di PriceDataBean:
 - 0 = non viene calcolata
 - 1 = viene calcolata

Il valore di PriceDataBean può essere utilizzato per ottenere gli importi delle tasse che devono essere mostrati insieme al prezzo del prodotto.

- b. Ripetere il passo a per ciascuna categoria di tasse utilizzata nel negozio.
- c. Utilizzando il seguente esempio come guida, aggiungere le informazioni sulla descrizione della categoria di tasse nel file XML per la tabella TAXCGRYDS. Se il negozio da creare è multiculturale, è necessario includere queste informazioni in un file XML specifico delle locale.

```

<taxcgrids
  taxcgr_id="@taxcgr_id_1"
  description="Sales Tax"
  language_id="en_US"/>

```

dove

- taxcgr_id è la categoria di tasse.
- description è una breve descrizione della categoria di tasse, che può essere visualizzata dai clienti.
- language_id è la lingua in cui vengono visualizzate queste informazioni.

- d. Ripetere il passo c per ciascuna categoria di tasse utilizzata nel negozio.

7. Definire i codici di calcolo utilizzati dal negozio.

- a. Utilizzando i seguenti esempi come guida, definire il codice di calcolo nel file XML per la tabella CALCODE.

```

<calcode
  calcode_id="@calcode_id_3"
  code="Tax Code 1"
  calusage_id="-3"
  storeent_id="@storeent_id_1"
  groupby="0"
  published="1"
  sequence="0"
  calmethod_id="-43"
  calmethod_id_app="-44"
  calmethod_id_qfy="-42"

```

```
displaylevel="0"
flags="0"
precedence="0"
/>
```

dove:

- calcode_id è una chiave univoca generata.
- code è una stringa di caratteri che identifica in modo univoco questo codice di calcolo, dati un particolare utilizzo di calcolo e un'entità negozio.
- calusage_id indica il tipo di calcolo per cui viene utilizzato questo codice. Ad esempio, il codice di calcolo può essere utilizzato per calcolare uno dei seguenti importi monetari:
 - Sconti (-1)
 - Spese di spedizione (-2)
 - Tasse di vendita (-3)
 - Tasse di spedizione (-4)
 - Coupon (-5)
- storeent_id è l'entità negozio associata a questo codice di calcolo.
- groupby sono indicatori che mostrano al metodo di combinazione codici di calcolo come raggruppare gli articoli dell'ordine durante l'esecuzione dei calcoli. Zero specifica nessun raggruppamento (tutti gli articoli dell'ordine applicabili fanno parte di un unico gruppo). Per ulteriori informazioni, fare riferimento a *tabella CALCODE: dettagli* nella guida in linea di WebSphere Commerce.
- published specifica se il codice di calcolo è calcolato:
 - 0 = non pubblicato (temporaneamente disabilitato)
 - 1 = pubblicato
 - 2 = contrassegnato per la cancellazione (e non pubblicato)
- sequence è l'ordine in cui viene calcolato il codice di calcolo. I codici di calcolo vengono calcolati e applicati in sequenza dal più basso al più alto. Se due codici di calcolo hanno lo stesso numero di sequenza, verranno calcolati per primi i codici di calcolo con il valore calcode_id più basso.
- calmethod_id è il metodo di calcolo del codice di calcolo che definisce come calcolare l'importo delle tasse per questo codice di calcolo. Per individuare il metodo di calcolo da utilizzare, attenersi alla seguente procedura:
 - Fare riferimento alla tabella CALMETHOD nella guida in linea di WebSphere Commerce. Fare riferimento alla descrizione per la colonna SUBCLASS. Fare clic sul collegamento alla tabella CALMETHOD: *dettagli*. Questa tabella elenca i tipi di CALMETHOD disponibili. Il metodo di calcolo è 3.
 - Aprire il file bootstrap wcs.bootstrap_xx_XX.xml, dove xx_XX è il codice della locale. I file bootstrap si trovano nelle seguenti directory:
 -  NT unità:\WebSphere\CommerceServer\schema\xml
 -  2000 unità:\Program Files\WebSphere\CommerceServer\schema\xml
 -  AIX /usr/WebSphere/CommerceServer/schema/xml
 -  Solaris /opt/WebSphere/CommerceServer/schema/xml

-  /opt/WebSphere/CommerceServer/schema/xml
-  /qibm/proddata/WebCommerce/schema/xml
- Individuare la sezione contenente i metodi di calcolo disponibili (CALMETHOD).
- Individuare i metodi di calcolo con il valore calusage_ID per le tasse (-3 per le tasse di vendita e -4 per le tasse di acquisto).
- Individuare il metodo di calcolo con la sottoclasse 3. Questo metodo di calcolo è -43.
- calmethod_id_app è il metodo di applicazione del codice di calcolo che memorizza l'importo calcolato per gli articoli associati. Utilizzare il metodo descritto in calmethod_id per stabilire il metodo di applicazione del codice di calcolo da utilizzare.
 - calmethod_id_app="-44" è il metodo di applicazione del codice di calcolo per le tasse di vendita
- calmethod_id_qfy è il metodo di qualifica del codice di calcolo che definisce gli articoli dell'ordine associati a questo codice di calcolo. Utilizzare il metodo descritto in calmethod_id per stabilire il metodo di qualifica del codice di calcolo da utilizzare.
 - calmethod_id_qfy="-42" è il metodo di qualifica del codice di calcolo per le tasse di vendita.
- display_level determina se gli importi calcolati da questo codice di calcolo devono essere visualizzati con:
 - 0 = articolo ordine
 - 1 = ordine
 - 2 = prodotto
 - 3 = articolo
 - 4 = contratto
- flags specifica se è necessario richiamare il metodo di qualifica del codice di calcolo di questo codice di calcolo.
 - 0 = illimitato. Il metodo non sarà richiamato
 - 1 = limitato. Il metodo sarà richiamato

b. Utilizzando il seguente esempio come guida, aggiungere le informazioni sulla descrizione del codice di calcolo nel file XML per la tabella CALCODEDSC. Se il negozio da creare è multiculturale, è necessario includere queste informazioni in un file XML specifico delle locale.

```
<calcodedsc
  calcode_id="@calcode_id_3"
  description="Vitamins
  language_id="&en_US"
  longdescription="In Ontario vitamins are taxed federally, but
  not provincially."
/>
```

dove

- calcode_id è il codice di calcolo a cui vengono applicate queste informazioni.
 - description è una descrizione breve del codice di calcolo.
 - language_id è la lingua in cui vengono applicate queste informazioni.
 - longdescription è la descrizione dettagliata del codice di calcolo.
- c. Ripetere i passi a e b per ciascun codice di calcolo utilizzato nel negozio.

8. Definire le regole di calcolo per il negozio.
- a. Utilizzando il seguente esempio come guida, impostare la regola di calcolo nel file XML per la tabella CALRULE:

```
<calrule
  calrule_id="@calrule_id_10"
  calcode_id="@calcode_id_3"
  startdate="1900-01-01 00:00:00.000000"
  taxcgry_id="@taxcgry_id_1"
  enddate="2100-01-01 00:00:00.000000"
  flags="1"
  identifier="1"
  combination="2"
  calmethod_id="-47"
  calmethod_id_qfy="-46"
/>
```

dove

- `calrule_id` è un identificativo univoco generato.
- `calcode_id` è il codice di calcolo di cui fa parte questa regola di calcolo.
- `startdate` è l'ora in cui questa regola di calcolo diventa effettiva.
- `taxcgry_id` è la categoria di tasse per cui è effettiva questa regola di calcolo.
- `enddate` è l'ora in cui questa regola di calcolo non è più effettiva.
- `combination` viene utilizzato dal metodo di combinazione regole di calcolo per determinare il modo in cui questa regola di calcolo può essere combinata con altre regole di calcolo. Per ulteriori informazioni, fare riferimento alla tabella CALRULE.
- `identifier` identifica questa regola di calcolo, insieme al relativo codice di calcolo.
- `flags` specifica l'indicatore che mostra l'elaborazione speciale che deve essere eseguita dall'implementazione metodo di combinazione regole di calcolo predefinita. Per ulteriori informazioni, fare riferimento alla tabella CALRULE nella guida in linea di WebSphere Commerce.
- `calmethod_id` è il metodo di calcolo regole di calcolo mediante il quale viene calcolato l'importo per un insieme di articoli dell'ordine. Per individuare il metodo di calcolo da utilizzare, attenersi alla seguente procedura:
 - Fare riferimento alla tabella CALMETHOD nella guida in linea di WebSphere Commerce. Fare riferimento alla descrizione per la colonna SUBCLASS. Fare clic sul collegamento alla tabella CALMETHOD: dettagli. Questa tabella elenca i tipi di CALMETHOD disponibili. Il metodo di calcolo è il numero 7.
 - Aprire il file `bootstrap wcs.bootstrap_xx_XX.xml`, dove `xx_XX` è il codice della locale. I file bootstrap si trovano nelle seguenti directory:
 -  `NT` unità:\WebSphere\CommerceServer\schema\xml
 -  `2000` unità:\Program Files\WebSphere\CommerceServer\schema\xml
 -  `AIX` /usr/WebSphere/CommerceServer/schema/xml
 -  `Solaris` /opt/WebSphere/CommerceServer/schema/xml
 -  `Linux` /opt/WebSphere/CommerceServer/schema/xml
 -  `400` /qibm/proddata/WebCommerce/schema/xml

- Individuare la sezione contenente i metodi di calcolo disponibili (CALMETHOD).
 - Individuare i metodi di calcolo con il valore calusage_ID per le tasse (-3 per le tasse di vendita e -4 per le tasse di acquisto).
 - Individuare il metodo di calcolo con la sottoclasse 7. Questo metodo di calcolo è -47.
 - calmethod_id_qfy è il metodo di qualifica regole di calcolo che determina l'insieme di articoli da inviare al metodo di calcolo regole di calcolo. Utilizzare il metodo descritto in calmethod_id per stabilire il metodo di qualifica della regola di calcolo da utilizzare.
- b. Ripetere il passo a per ciascuna regola di calcolo utilizzata nel negozio. Ciascun codice di calcolo può avere diverse regole di calcolo, una per ciascuna categoria di tasse applicabile. Ad esempio, calcode_id="@calcode_id_1" può essere associato a diversi calrule_id.
9. Definire la scala di calcolo per il negozio.

Una scala di calcolo è l'insieme di intervalli che si applicano al calcolo. Questi intervalli creano una scala.

- a. Utilizzando il seguente esempio come guida, impostare la scala di calcolo nel file XML per la tabella CALSCALE:

```
<calscale
  calscale_id="@calscale_id_19"
  code="Sales Tax 1"
  storeent_id="@storeent_id_1"
  calusage_id="-3"
  setccurr="USD"
  calmethod_id="-53"
/>
```

dove

- calscale_id è un identificativo univoco generato.
- code è una stringa di caratteri che identifica in modo univoco questa scala di calcolo, dati un particolare utilizzo di calcolo e un'entità negozio.
- storeent_id è l'entità negozio di cui fa parte questa scala di calcolo.
- calusage_id indica il tipo di calcolo per cui viene utilizzata questa scala di calcolo. Ad esempio, la scala di calcolo può essere utilizzata per calcolare uno dei seguenti importi monetari:
 - sconti (-1)
 - spese di spedizione (-2)
 - tasse di vendita (-3)
 - tasse di spedizione (-4)
 - coupon (-5)
- setccurr se specificato, indica la valuta per i valori iniziali dell'intervallo degli oggetti di calcolo per questa regola di calcolo. Il metodo di ricerca della scala di calcolo restituisce un "numero di ricerca" in questa valuta. In questo caso, il valore non è specificato; il metodo di ricerca della scala di calcolo restituisce un numero di ricerca nella valuta dell'ordine. La valuta non deve essere specificata, a meno che i valori di inizio dell'intervallo della scala non siano diversi da zero.
- calmethod_id è il metodo di ricerca della scala di calcolo che, dato un insieme di articoli dell'ordine, determina un numero di ricerca, un valore monetario di base, un moltiplicatore di risultato ed un insieme di misure matematiche che possono essere utilizzate dalla scala di calcolo per

calcolare un importo. Per individuare il metodo di ricerca della scala di calcolo da utilizzare, attenersi alla seguente procedura:

- Fare riferimento alla tabella CALMETHOD nella guida in linea di WebSphere Commerce. Fare riferimento alla descrizione per la colonna SUBCLASS. Fare clic sul collegamento alla tabella CALMETHOD: dettagli. Questa tabella elenca i tipi di CALMETHOD disponibili. Il metodo di ricerca della scala di calcolo monetario è il numero 9.
- Aprire il file bootstrap wcs.bootstrap_xx_XX.xml, dove xx_XX è il codice della locale. I file bootstrap si trovano nelle seguenti directory:
 -  NT unità:\WebSphere\CommerceServer\schema\xml
 -  2000 unità:\Program Files\WebSphere\CommerceServer\schema\xml
 -  AIX /usr/WebSphere/CommerceServer/schema/xml
 -  Solaris /opt/WebSphere/CommerceServer/schema/xml
 -  Linux /opt/WebSphere/CommerceServer/schema/xml
 -  400 /qibm/proddata/WebCommerce/schema/xml
- Individuare la sezione contenente i metodi di calcolo disponibili (CALMETHOD).
- Individuare i metodi di calcolo con il valore calusage_ID per le tasse (-3 per le tasse di vendita e -4 per le tasse di acquisto).
- Individuare il metodo di calcolo con la sottoclasse 9. Esistono vari metodi di calcolo con la sottoclasse 9. Selezionare quello che risponde alle proprie esigenze.

Per ulteriori informazioni, fare riferimento alla tabella CALSCALE nella guida in linea di WebSphere Commerce.

- b. Ripetere il passo a per ciascuna scala di calcolo utilizzata nel negozio.
- c. Utilizzando il seguente esempio come guida, aggiungere le informazioni sulla descrizione della scala di calcolo nel file XML per la tabella CALSCALDS. Se il negozio da creare è multiculturale, è necessario includere queste informazioni in un file XML specifico delle locale.

```
<calscaleds
  calscale_id="@calscale_id_19"
  description="Sales Tax 5%"
  language_id="&en_US"
/>
```

dove

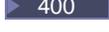
- calscale_id è la scala di calcolo a cui viene applicata questa descrizione.
 - description è una breve descrizione della scala di calcolo, che può essere visualizzata a un cliente per spiegare come viene eseguito il calcolo. Ad esempio, "\$.10 al chilo, tassa minima \$5.00." o "10% di sconto per quantità da 5 in su".
 - language_id è la lingua in cui vengono visualizzate queste informazioni.
- d. Ripetere il passo c per ciascuna scala di calcolo utilizzata nel negozio.
10. Definire gli intervalli di calcolo per le scale di calcolo.
- a. Utilizzando il seguente esempio come guida, impostare l'intervallo di calcolo nel file XML per la tabella CALRANGE.

```

<calrange
  calrange_id="@calrange_id_37"
  calscale_id="@calscale_id_19"
  calmethod_id="-59"
  rangestart="0.00000"
  cumulative="0"
/>

```

dove

- calrange_id è un identificativo univoco generato.
- calscale_id è la scala di calcolo di cui fa parte questo intervallo di calcolo.
- calmethod_id è il metodo di intervallo del calcolo che restituisce un importo monetario come risultato della ricerca di intervallo del calcolo. Ad esempio, FixedAmountCalculationRangeCmd, PerUnitAmountCalculationRangeCmd oppure PercentageCalculationRangeCmd. Per stabilire il metodo di intervallo del calcolo, attenersi alla seguente procedura:
 - Fare riferimento alla tabella CALMETHOD nella guida in linea di WebSphere Commerce. Fare riferimento alla descrizione per la colonna SUBCLASS. Fare clic sul collegamento alla tabella CALMETHOD: dettagli. Questa tabella elenca i tipi di CALMETHOD disponibili. Il metodo di intervallo del calcolo è il numero 10.
 - Aprire il file bootstrap wcs.bootstrap_xx_XX.xml, dove xx_XX è il codice della locale. I file bootstrap si trovano nelle seguenti directory:
 -  NT unità:\WebSphere\CommerceServer\schema\xml
 -  2000 unità:\Program Files\WebSphere\CommerceServer\schema\xml
 -  AIX /usr/WebSphere/CommerceServer/schema/xml
 -  Solaris /opt/WebSphere/CommerceServer/schema/xml
 -  Linux /opt/WebSphere/CommerceServer/schema/xml
 -  400 /qibm/proddata/WebCommerce/schema/xml
 - Individuare la sezione contenente i metodi di calcolo disponibili (CALMETHOD).
 - Individuare i metodi di calcolo con il valore calusage_ID per le tasse (-3 per le tasse di vendita e -4 per le tasse di acquisto).
 - Individuare il metodo di calcolo con la sottoclasse 10. Esistono diversi metodi di calcolo con la sottoclasse 10. Selezionare il metodo di calcolo che risponde alle proprie necessità.
- rangestart: se un numero di ricerca è maggiore o uguale a RANGESTART oppure se RANGESTART è NULL, questa riga corrisponde al numero di ricerca.
- cumulative è pari a quanto di seguito riportato:
 - 0 = viene utilizzato solo l'intervallo di calcolo corrispondente con il valore RANGESTART più elevato.
 - 1 = vengono utilizzati tutti gli intervalli di calcolo corrispondenti. Gli importi calcolati vengono sommati in modo da ottenere il risultato finale.

Per ulteriori informazioni, fare riferimento alla tabella CALRANGE nella guida in linea di WebSphere Commerce.

- b. Ripetere il passo a per ciascun intervallo di calcolo associato alla scala di calcolo utilizzata nel negozio. Nell'esempio sopra, esiste un unico intervallo, dal momento che tutti gli importi sono tassati allo stesso modo.
11. Definire i valori di ricerca del calcolo per le scale di calcolo. I valori di ricerca del calcolo sono i valori associati alla scala di calcolo. Ad esempio, una scala di calcolo include i seguenti intervalli e tasse associate per le tasse di vendita provinciali dell'Ontario sul cibo servito in un ristorante:

- \$0.00 - \$3.99 tassati allo 0.00%
- \$4.00 in su tassati all'8.00%

I valori di ricerca sono 0.00 e 8.00.

- a. Utilizzando i seguenti esempi come guida, impostare la ricerca di calcolo nel file XML per la tabella CALRLOOKUP.

```
<calrlookup
  calrlookup_id="@calrlookup_id_37"
  calrange_id="@calrange_id_37"
  value="5.00"
/>
```

dove

- calrlookup_id è un identificativo univoco generato.
- calrange_id è l'intervallo di calcolo di cui fa parte il risultato della ricerca dell'intervallo di calcolo.
- value è il valore del risultato della ricerca dell'intervallo di calcolo, utilizzato dal metodo dell'intervallo di calcolo per stabilire un importo. In questo esempio la tassa è del 5%.

Per ulteriori informazioni, fare riferimento alla tabella CALRLOOKUP nella guida in linea di WebSphere Commerce.

- b. Ripetere i passi a e b per ciascun valore di ricerca associato alla scala di calcolo utilizzata nel negozio. In questo esempio, esiste un unico valore CALRLOOKUP, dal momento che CALRLOOKUP.SETCCURR è NULL ed esiste un unico valore CALRANGE, dal momento che la tassa è la stessa per tutti gli importi.
12. Associare la regola di calcolo e la scala di calcolo.

- a. Utilizzando i seguenti esempi come guida, associare la scala di calcolo alla regola di calcolo nel file XML per la tabella CRULESCALE.

```
<crulescale
  calrule_id="@calrule_id_10"
  calscale_id="@calscale_id_19"
/>
```

dove

- calrule_id è la regola di calcolo.
 - calscale_id è la scala di calcolo.
- b. Ripetere il passo a per ciascuna associazione di scala di calcolo e regola. Nell'esempio utilizzato sopra, esiste un'unica scala di calcolo per ciascuna regola di calcolo.

Nota: Se la tassa varia in base alla quantità acquistata, è necessario creare scale con valori rangestart diversi da zero. Quindi, è necessario creare una scala di calcolo per ciascuna valuta supportata (impostando CALSCALE.SETCCURR sulla valuta appropriata) per la quale non è stato stabilito un tasso di conversione (fare riferimento alla tabella CURCONVERT) e associare i valori alla

regola di calcolo per quella particolare categoria di tasse. Ad esempio, non esiste alcuna tassa provinciale nell'Ontario sul cibo al di sotto di \$4.00. Se il negozio supporta la vendita di cibo in dollari americani, è necessario stabilire una conversione dai dollari americani ai dollari canadesi oppure creare una scala di calcolo delle tasse diversa con un valore rangestart appropriato, ad esempio \$6.00 USD e associare tale valore alla stessa regola di calcolo. Verrà utilizzata solo la scala di calcolo appropriata, in base alla valuta dell'ordine.



Per ulteriori informazioni sull'uso di @ e &, fare riferimento all'Appendice B, "Creazione di dati" a pagina 319.

Creazione di risorse evasione ordini per le tasse

Per fare in modo che le risorse tasse funzionino correttamente nel negozio, è necessario associare i gruppi di giurisdizioni di tasse al centro di evasione ordini utilizzato dal negozio e associare a entrambi una regola di calcolo.

Per poter associare le risorse tasse ad un centro di evasione ordini, è necessario creare le risorse evasione ordini. Per ulteriori informazioni sulla creazione delle risorse evasione ordini, fare riferimento alla sezione "Creazione di risorse evasione ordini in WebSphere Commerce" a pagina 112.

Una volta create le risorse evasione ordini, associarvi le risorse tasse, aggiungendo informazioni alla tabella TAXJCRULE. Attenersi alla procedura di seguito riportata:

1. Rivedere il manuale *IBM WebSphere Commerce Calculation Framework Guide*. Il framework di calcolo di WebSphere Commerce calcola gli importi monetari (ad esempio quello relativo alle tasse) associati al prodotto o servizio scelto per l'acquisto da un cliente.
2. Rivedere i file XML utilizzati per creare le risorse evasione ordini relative alle tasse per i negozi di esempio. Tutti i file per i negozi di esempio si trovano nel file dell'archivio negozio corrispondente.

I file dell'archivio negozio si trovano nelle seguenti directory:

-  NT unità:\WebSphere\CommerceServer\samplstores
-  2000 unità:\Program Files\WebSphere\CommerceServer\samplstores
-  AIX /usr/WebSphere/CommerceServer/samplstores
-  Solaris /opt/WebSphere/CommerceServer/samplstores
-  Linux /opt/WebSphere/CommerceServer/samplstores
-  400 /qibm/proddata/WebCommerce/samplstores

Nota: La guida in linea di WebSphere Commerce contiene le informazioni su ciascuna delle risorse dati contenute nei negozi di esempio.

Ciascun negozio di esempio presenta un file `taxfulfill.xml`, contenente le informazioni sulle tasse. Per visualizzare il file `taxfulfill.xml` nell'archivio negozio, decomprimere il file con un programma ZIP. Il file `taxfulfill.xml` si trova nella directory dei dati.

3. Fare riferimento alle informazioni riportate in Appendice B, "Creazione di dati" a pagina 319.

4. Creare un file `taxfulfill.xml`, copiando uno dei file `taxfulfill.xml` presenti negli archivi negozio di esempio o creandone uno nuovo. Per ulteriori informazioni, fare riferimento al file DTD corrispondente a `taxfulfill.xml`. I file DTD si trovano nelle seguenti directory:

- ▶ **NT** unità:\WebSphere\CommerceServer\xml\sar
- ▶ **2000** unità:\Program Files\WebSphere\CommerceServer\xml\sar
- ▶ **AIX** /usr/WebSphere/CommerceServer/xml/sar
- ▶ **Solaris** /opt/WebSphere/CommerceServer/xml/sar
- ▶ **Linux** /opt/WebSphere/CommerceServer/xml/sar
- ▶ **400** /qibm/proddata/WebCommerce/xml/sar

5. Utilizzando il seguente esempio come guida, nel file XML aggiungere informazioni per la tabella TAXJCRULE:

```
<taxjcrule
taxjcrule_id="@taxjcrule_id_1"
calrule_id="@calrule_id_10"
ffmcenter_id="@ffmcenter_id_1"
jurstgroup_id="@jurstgroup_id_2"
precedence="0"
/>
```

dove

- `taxjcrule_id` è un identificativo univoco generato.
 - `calrule_id` è la regola di calcolo utilizzata.
 - `ffmcenter_id` è il centro di evasione ordini. Se è NULL, questa associazione si applica a tutti i centri di evasione ordini.
 - `jurstgroup_id` è il gruppo di giurisdizioni di tasse. Se è NULL, questa associazione si applica a tutti i gruppi di giurisdizioni di tasse.
 - `precedence` è il valore di precedenza relativo alle regole di calcolo. Quando un indirizzo di spedizione è compreso in più gruppi di giurisdizioni di tasse specificati, per lo stesso centro di evasione ordini, viene qualificata solo la regola di calcolo con il valore TAXJCRULE.PRECEDENCE maggiore.
6. Ripetere il passo 3 per ciascuna associazione tra gruppi di giurisdizioni, centri di evasione ordini e regole.



Per ulteriori informazioni sull'uso di @ e &, fare riferimento all'Appendice B, "Creazione di dati" a pagina 319.

Creazione di risorse tasse catalogo del negozio

Per associare le tasse alle merci e ai servizi del negozio, è necessario associare un codice di calcolo alle voci di catalogo del negozio, per ciascun contratto del negozio.

Per poter creare le risorse tasse catalogo del negozio, è necessario creare le risorse del negozio e del catalogo. Per ulteriori informazioni sulla creazione delle risorse del negozio, fare riferimento alla sezione "Creazione di risorse dati del negozio in un file XML" a pagina 45. Per ulteriori informazioni sulla creazione delle risorse catalogo, consultare la sezione "Visualizzazione delle risorse catalogo del negozio" a pagina 77.

Per creare le risorse tasse catalogo del negozio, procedere come segue:

1. Rivedere il manuale *IBM WebSphere Commerce Calculation Framework Guide*. Il framework di calcolo di WebSphere Commerce calcola gli importi monetari (ad esempio quello relativo alla spedizione) associati al prodotto o servizio scelto per l'acquisto da un cliente.
2. Rivedere i file XML utilizzati per creare le risorse tasse catalogo del negozio per i negozi di esempio. Tutti i file per i negozi di esempio si trovano nel file dell'archivio negozio corrispondente.

I file dell'archivio negozio si trovano nelle seguenti directory:

-  NT unità:\WebSphere\CommerceServer\samplstores
-  2000 unità:\Program Files\WebSphere\CommerceServer\samplstores
-  AIX /usr/WebSphere/CommerceServer/samplstores
-  Solaris /opt/WebSphere/CommerceServer/samplstores
-  Linux /opt/WebSphere/CommerceServer/samplstores
-  400 /qibm/proddata/WebCommerce/samplstores

Ogni negozio di esempio presenta un file `store-catalog-tax.xml`, che comprende informazioni sull'evasione ordini relativa alla spedizione. Per visualizzare il file `store-catalog-tax.xml` nell'archivio negozio, decomprimere il file con un programma ZIP. Il file `store-catalog-tax.xml` si trova nella directory dei dati.

Nota: La guida in linea di WebSphere Commerce contiene le informazioni su ciascuna delle risorse dati contenute nei negozi di esempio.

3. Fare riferimento alle informazioni riportate in Appendice B, "Creazione di dati" a pagina 319.
4. Creare un file `store-catalog-tax.xml`, copiando uno dei file `store-catalog-tax.xml` presenti negli archivi negozio di esempio o creandone uno nuovo. Per ulteriori informazioni, fare riferimento al file DTD corrispondente a `store-catalog-tax.xml`. I file DTD si trovano nelle seguenti directory:

-  NT unità:\WebSphere\CommerceServer\xml\sar
-  2000 unità:\Program Files\WebSphere\CommerceServer\xml\sar
-  AIX /usr/WebSphere/CommerceServer/xml/sar
-  Solaris /opt/WebSphere/CommerceServer/xml/sar
-  Linux /opt/WebSphere/CommerceServer/xml/sar
-  400 /qibm/proddata/WebCommerce/xml/sar

5. Creare la relazione tasse catalogo del negozio aggiungendo informazioni alla tabella `CATENCALCD`. Utilizzare il seguente esempio come guida:

```
<catenacalcd
  calcode_id="@calcode_id_3"
  catenacalcd_id="@catenacalcd_id_3"
  store_id="@storeent_id_1"
/>
```

dove

- `calcode_id` è il codice di calcolo.
- `catenacalcd_id` è un identificativo univoco generato.

- `store_id` è il negozio.



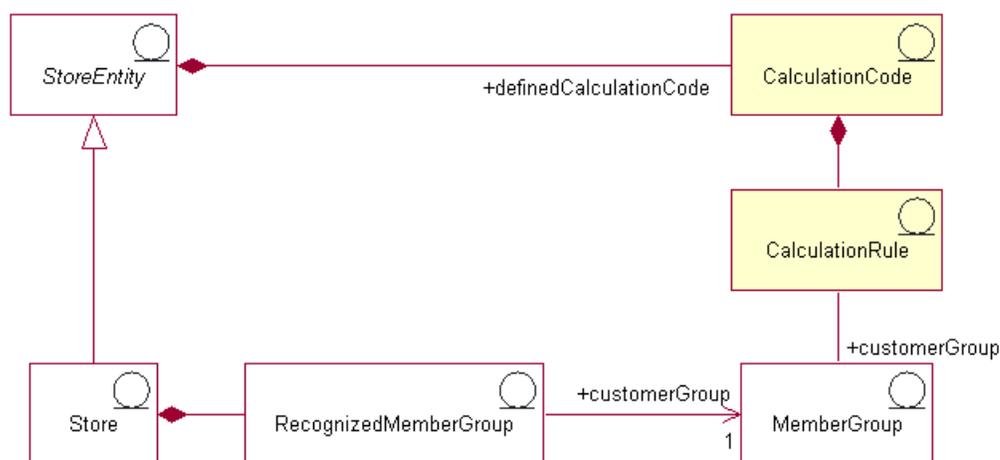
Per ulteriori informazioni sull'uso di @ e &, fare riferimento all'Appendice B, "Creazione di dati" a pagina 319.

Capitolo 20. Risorse sconti

Gli sconti offrono ai clienti incentivi per gli acquisti. E' possibile offrire sconti in percentuale (come 10% in meno) o sconti a quantità fissa (come 15 euro in meno). Gli sconti possono essere applicati a singoli prodotti oppure alla somma totale degli acquisti. Ad esempio, è possibile offrire una riduzione del 20% ai cittadini adulti oppure, se si dispone di molti berretti rossi da baseball in magazzino, è possibile offrire uno sconto del 25% sui cappellini per un periodo di tempo limitato.

Sconti in WebSphere Commerce

Il grafico di seguito riportato illustra la struttura degli sconti in WebSphere Commerce Server.



Questo grafico, insieme a tutti gli altri grafici della sezione relativa ai dati del negozio fa parte del modello di informazioni di WebSphere Commerce Server. Per ulteriori informazioni sul modello di informazioni, fare riferimento alla sezione "Modello di informazioni dati del negozio" a pagina 27. Per ulteriori informazioni sulle convenzioni utilizzate in questo grafico, fare riferimento all'Appendice A, "Legenda UML" a pagina 317.

Codice di calcolo

Uno *sconto* viene rappresentato e calcolato utilizzando un *codice di calcolo* degli sconti. Un codice di calcolo degli sconti indica il modo in cui lo sconto viene calcolato per gli articoli dell'ordine.

Un codice di calcolo è parte di un'entità negozio. All'interno di un'entità negozio possono essere definiti più codici di calcolo. Se l'entità negozio viene cancellata, vengono cancellati anche i codici di calcolo definiti con l'entità negozio.

Ciascun codice di calcolo degli sconti può avere una data iniziale e una data finale che definiscono il periodo di tempo in cui lo sconto è effettivo. Il codice di calcolo degli sconti può essere anche associato ad uno o più gruppi di membri, che definiscono i gruppi di membri idonei.

Il codice di calcolo degli sconti può essere assegnato a una o più voci di catalogo e gruppi di catalogo. L'assegnazione di un codice di calcolo a un gruppo di catalogo equivale ad assegnare il codice direttamente a tutte le voci di catalogo del gruppo. Tuttavia, i codici di calcolo degli sconti collegati al gruppo di catalogo A non sono collegati ai prodotti e agli articoli del gruppo di catalogo B se il gruppo di catalogo A contiene il gruppo di catalogo B.

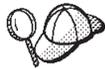
Alle voci di catalogo o ai gruppi di catalogo possono essere associati più sconti. Se a un ordine è applicabile più di un codice di calcolo di sconto, i calcoli vengono eseguiti nella sequenza ascendente degli attributi di sequenza del codice di calcolo.

Nota: Definire gli ordini di sequenza degli sconti per applicare ulteriori sconti agli sconti.

Gli articoli dell'ordine sono raggruppati per il calcolo in uno dei seguenti modi:

- Per accordo commerciale
- Per prodotto
- Per offerta
- Per indirizzo di spedizione

Per ulteriori informazioni, fare riferimento alla guida in linea di WebSphere Commerce.



Per ulteriori informazioni sull'utilizzo dei codici di calcolo, consultare il manuale *IBM WebSphere Commerce Calculation Framework Guide*.

Regole di calcolo

Ciascun codice di calcolo dispone di un insieme di regole di calcolo che definiscono le condizioni per effettuare il calcolo. Ciascuna regola di calcolo degli sconti è associata ad uno o più gruppi di membri, per il quale lo sconto è valido. I gruppi di membri possono essere idonei a ricevere più di uno sconto alla volta.

Nota: Se un gruppo di membri idoneo è definito a livello di codice di calcolo, non deve essere definito anche a livello di regola di calcolo.



Per ulteriori informazioni sull'utilizzo delle regole di calcolo, consultare il manuale *IBM WebSphere Commerce Calculation Framework Guide*.

Creazione di risorse sconti in WebSphere Commerce

Il metodo principale per creare sconti in un negozio generato con WebSphere Commerce, consiste nell'utilizzare la procedura guidata Sconto in WebSphere Commerce Accelerator. Per ulteriori informazioni sulla creazione degli sconti mediante WebSphere Commerce Accelerator, fare riferimento alla guida in linea di WebSphere Commerce.

Gli sconti possono essere creati anche mediante un file XML, quindi caricati con il pacchetto Loader o pubblicati mediante Servizi del negozio. Tuttavia, gli sconti creati in questo modo o importati durante la migrazione da una precedente versione, funzioneranno correttamente ma non verranno visualizzati correttamente in WebSphere Commerce Accelerator.

Capitolo 21. Risorse inventario

L'inventario include qualsiasi cosa possa essere considerata fisicamente in un centro di evasione ordini. Esistono specifiche definizioni per i tipi di inventario, quali articoli, prodotti, SKU, bundle e pacchetti; ma tali elementi sono tutti considerati come inventario. I prodotti sono configurati per l'evasione ordini nella procedura guidata Prodotto e nel blocco appunti Prodotto. Il blocco appunti Prodotto include opzioni per tenere traccia dell'inventario, consentire sospesi, imporre sospesi, rilasciare i prodotti separatamente e specificare che il prodotto non può essere restituito. WebSphere Commerce Accelerator fa una distinzione tra i due maggiori tipi di inventario che è possibile ricevere:

- Inventario previsto, a cui è associato un record di inventario previsto.
- Inventario ad hoc, o inventario non registrato come previsto.

L'inventario previsto viene ricevuto da un fornitore e generalmente pagato con un ordine di acquisto. WebSphere Commerce Accelerator tiene traccia dell'inventario atteso con i record dell'inventario atteso e consente di registrare un identificatore esterno, normalmente un numero dell'ordine di acquisto, da un sistema esterno. In questo modo, è possibile tenere traccia facilmente dell'inventario ordinato e di ciò che è arrivato o meno. I dettagli di inventario previsto sono specifiche sui prodotti in un record di inventario previsto, come il centro di evasione ordini in attesa del prodotto, la data di ricezione prevista, la quantità prevista e i commenti.

Un record di un inventario previsto non può essere eliminato una volta che l'inventario è stato ricevuto e i dettagli dell'inventario previsto non possono essere modificati o eliminati una volta che tutti quelli relativi all'inventario sono stati ricevuti.

Quando vengono effettuati gli ordini per l'inventario che è disponibile in un centro di evasione ordini, il sistema dell'ordine assegna l'inventario a quegli ordini. L'assegnazione dell'inventario ad un ordine lo rende non disponibile al sistema dell'ordine. Se l'ordine viene annullato, l'inventario è di nuovo disponibile. Se viene effettuato un ordine per l'inventario che non è disponibile, è possibile creare un ordine in sospeso. Se esiste un inventario previsto che potrebbe essere utilizzato per smistare l'ordine in sospeso, l'inventario viene assegnato a tale ordine ed è possibile fornire al cliente una data di spedizione prevista.

Le ricevute dell'inventario ad hoc vengono create quando l'inventario giunge ad un centro di evasione ordini senza un corrispondente record di inventario previsto. Ciò potrebbe essere dovuto ad un arrivo inaspettato dell'inventario o potrebbe dalla scelta di un venditore di non utilizzare record di inventario previsti per registrare le ricevute dell'inventario.

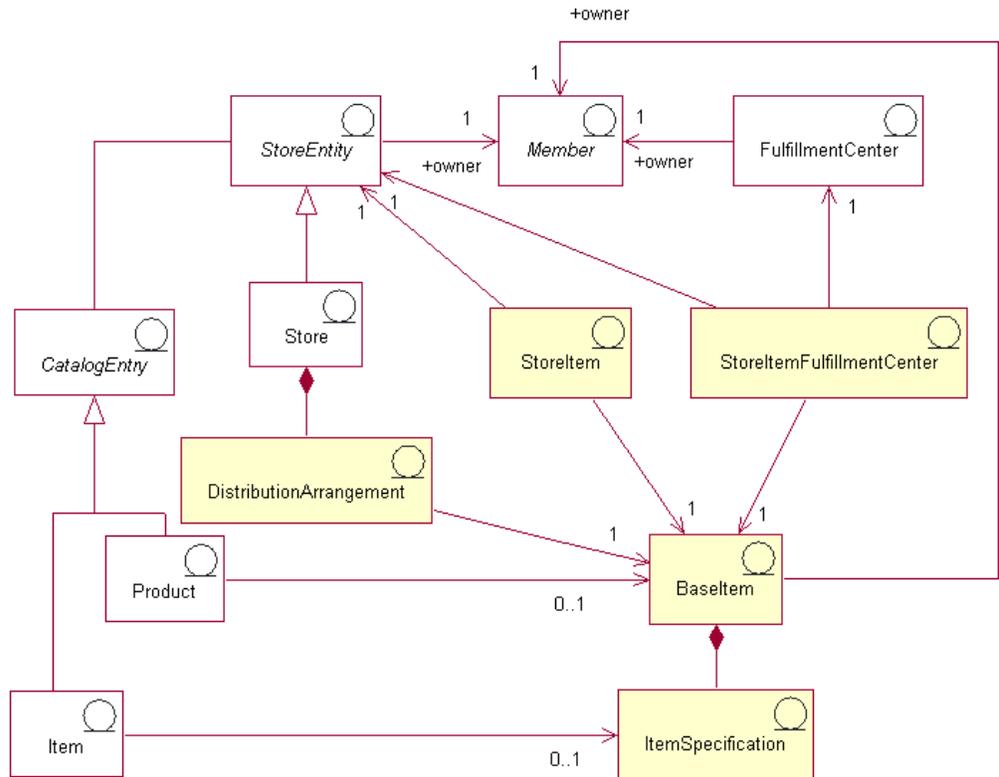
Nota: E' necessario che i prodotti siano presenti nel sistema WebSphere Commerce per essere ricevuti, se è la ricevuta di inventario è prevista o mirata.

Risorse inventario in WebSphere Commerce

Per comprendere le risorse inventario è necessario conoscere le relazioni tra l'inventario e il negozio. Tali relazioni possono essere spiegate utilizzando un modello di informazioni. Le sezioni di seguito riportate descrivono le relazioni e le associazioni tra un inventario, un negozio e altre risorse. I grafici di seguito

riportati illustrano le relazioni e le associazioni per l'inventario ATP e non ATP. L'inventario non ATP è il tipo di inventario gestito dalle versioni precedenti del prodotto e può essere ancora utilizzato se un negozio stabilisce di non utilizzare le funzioni ATP. I grafici e le relative associazioni sono descritti di seguito. Per ulteriori informazioni su ATP, fare riferimento alla guida in linea.

Inventario ATP



Questo grafico, insieme a tutti gli altri grafici della sezione relativa ai dati del negozio fa parte del modello di informazioni di WebSphere Commerce Server. Per ulteriori informazioni sul modello di informazioni, fare riferimento alla sezione "Modello di informazioni dati del negozio" a pagina 27. Per ulteriori informazioni sulle convenzioni utilizzate in questo grafico, fare riferimento all'Appendice A, "Legenda UML" a pagina 317.

Articolo di base

L'*articolo di base* è il centro del grafico dell'inventario e rappresenta un gruppo generale di merci con un nome e una descrizione comune. Gli articoli di base vengono utilizzati esclusivamente per l'evasione ordini e non sono particolari per ciascun negozio. Ogni voce di catalogo che rappresenta un prodotto del catalogo ha un articolo di base corrispondente per scopi di evasione ordini. Gli articoli di base sono definiti nella tabella BASEITEM.

Specifica articolo

La *specifica di un articolo* è un articolo di base con valori definiti per tutti gli attributi. Ogni voce di catalogo che rappresenta un articolo del catalogo ha una specifica di articolo corrispondente per scopi di evasione ordini.

Voci di catalogo

I prodotti e gli articoli sono *voci di catalogo*. Le voci di catalogo sono associate alle entità negozio, il che significa che le voci di catalogo, come prodotti e articoli, si trovano nei negozi.

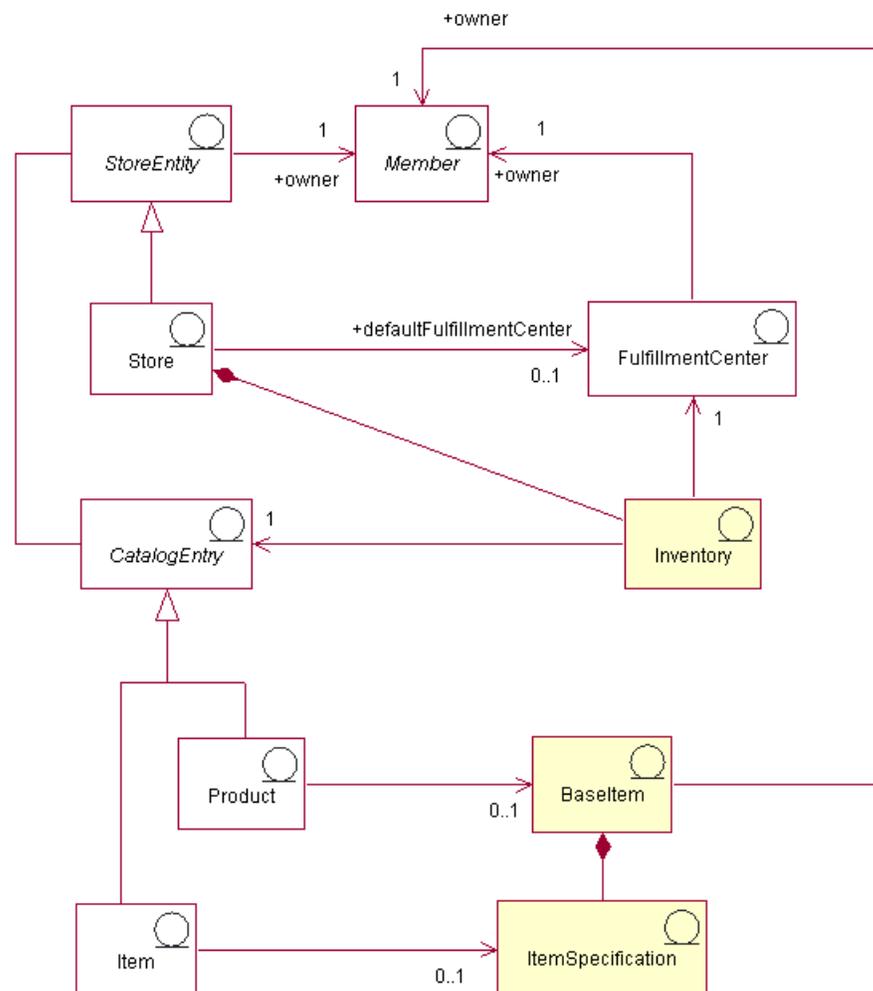
Disposizioni relative alla distribuzione

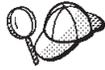
Una *disposizione relativa alla distribuzione* è associata ad un articolo di base, il che consente ad un negozio di vendere il relativo inventario. Le disposizioni relative alla distribuzione si trovano nella tabella DISTARRANG.

Articolo di negozio

Un *articolo di negozio* rappresenta gli attributi che determinano il modo in cui un determinato negozio o gruppo di negozi assegna l'inventario per gli articoli specificati di un determinato articolo di base, inclusi gli ordini in sospeso e l'inventario di traccia. Nella tabella STORITMFFC viene definita una stima del numero di secondi necessari dall'ora del rilascio di un articolo dell'ordine per l'evasione ordini fino al momento della sua spedizione al cliente. Questa tabella contiene delle voci solo se un negozio desidera definire una sovrapposizione per l'offset di spedizione predefinito FFMCCENTER per un articolo dell'ordine.

Inventario non ATP





Questo grafico, insieme a tutti gli altri grafici della sezione relativa ai dati del negozio fa parte del modello di informazioni di WebSphere Commerce Server. Per ulteriori informazioni sul modello di informazioni, fare riferimento alla sezione "Modello di informazioni dati del negozio" a pagina 27. Per ulteriori informazioni sulle convenzioni utilizzate in questo grafico, fare riferimento all'Appendice A, "Legenda UML" a pagina 317.

L'articolo di base è il centro del grafico dell'inventario non ATP. La relazione dell'articolo di base con prodotti, articoli e voci di catalogo è uguale a quella del grafico dell'inventario generale. Un articolo di base è sempre di proprietà di un membro e, una volta definito da quel membro, può essere venduto nel negozio. In questo caso, tuttavia, non esiste alcuna disposizione per la distribuzione, associazione di articoli del negozio o centro di evasione ordini per gli articoli del negozio.

Centro di evasione ordini

Ogni inventario è associato ad un unico *centro di evasione ordini*. Un negozio può scegliere un centro di evasione ordini predefinito. Come gli articoli di base, i centri di evasione ordini sono di proprietà dei membri e condividono quella proprietà con il negozio. Per ulteriori informazioni sulle risorse evasione ordini, fare riferimento al Capitolo 11, "Risorse evasione ordini" a pagina 109.



Per informazioni più dettagliate sulla struttura delle risorse inventario in WebSphere Commerce Server, fare riferimento ai modelli di oggetti e dati di inventario nella guida in linea di WebSphere Commerce.

Creazione di risorse inventario in WebSphere Commerce

L'inventario, essendo costituito da dati operativi, cambia ogni giorno, quando i clienti acquistano prodotti dal negozio o restituiscono i prodotti acquistati. I livelli dell'inventario, quindi, salgono e scendono in base alla vendita dei prodotti e in base ai nuovi inventari ricevuti dai centri di evasione ordini da parte dei fornitori. WebSphere Commerce Accelerator consente di completare le attività di seguito riportate relative all'inventario:

- Registrazione inventario previsto
- Ricezione inventario previsto e ad hoc da parte dei fornitori
- Adattamento inventario
- Gestione record restituzioni
- Gestione motivi restituzioni
- Ricezione inventari restituiti da parte dei clienti
- Gestione disposizione inventari restituiti

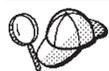
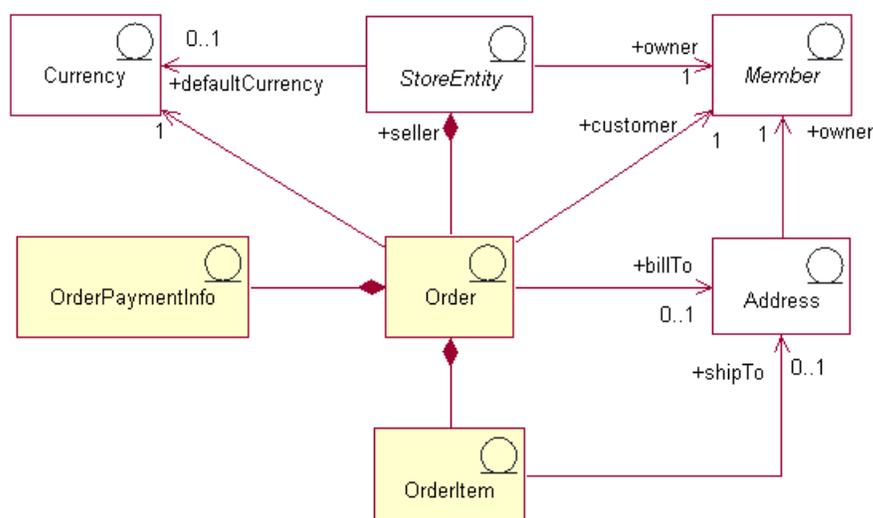
Per ulteriori informazioni sull'uso di WebSphere Commerce Accelerator per la gestione degli inventari, fare riferimento alla guida in linea di WebSphere Commerce.

Capitolo 22. Risorse ordini

Le risorse ordini nel sistema WebSphere Commerce forniscono le funzioni di carrello degli acquisti, gestione ordini e elaborazione ordini. Le funzioni di elaborazione ordini includono l'ordinazione o l'acquisto rapido, gli ordini pianificati, gli ordini in sospeso multipli, gli ordini ripetuti e gli ordini divisi o arretrati. I relativi servizi, come l'assegnazione dei prezzi, la tassazione, il pagamento, l'inventario e l'evasione ordini, sono parte delle risorse ordini.

Risorse ordini in WebSphere Commerce

Il grafico di seguito riportato illustra le risorse ordini in WebSphere Commerce Server. Dopo il grafico, è riportata la descrizione di ciascuna risorsa.



Questo grafico, insieme a tutti gli altri grafici della sezione relativa ai dati del negozio fa parte del modello di informazioni di WebSphere Commerce Server. Per ulteriori informazioni sul modello di informazioni, fare riferimento alla sezione "Modello di informazioni dati del negozio" a pagina 27.

Ordini e articoli degli ordini

Nel sistema WebSphere Commerce, per un cliente o acquirente, un *ordine* è un elenco di prodotti selezionati (ad esempio un ordine può contenere due libri e un CD) e ciascun prodotto dell'elenco è un *articolo dell'ordine* (ad esempio ciascun libro e il CD sono articoli dello stesso ordine). Quando un cliente effettua un ordine al negozio, deve fornire un indirizzo di fatturazione a cui il negozio deve inviare la fattura. A ciascun ordine è associato un solo identificativo della valuta. Dalla prospettiva del negozio, un ordine è un elenco di articoli dell'ordine. L'ordine fa parte dei dati del negozio.

Valuta

Un negozio può visualizzare i prezzi in un'unica *valuta* o utilizzare più valute. Ogni negozio deve inoltre definire una *valuta predefinita*. E' inoltre possibile consentire ai clienti di selezionare una *valuta di acquisto*. Se la valuta di acquisto corrisponde alla valuta predefinita per il negozio, essa è già supportata nella

tabella STOREENT. Se la valuta di acquisto non corrisponde alla valuta predefinita per il negozio, è necessario aggiungerla alla tabella CURLIST. I clienti utilizzano la valuta di acquisto per effettuare gli ordini al negozio.

Informazioni sul pagamento

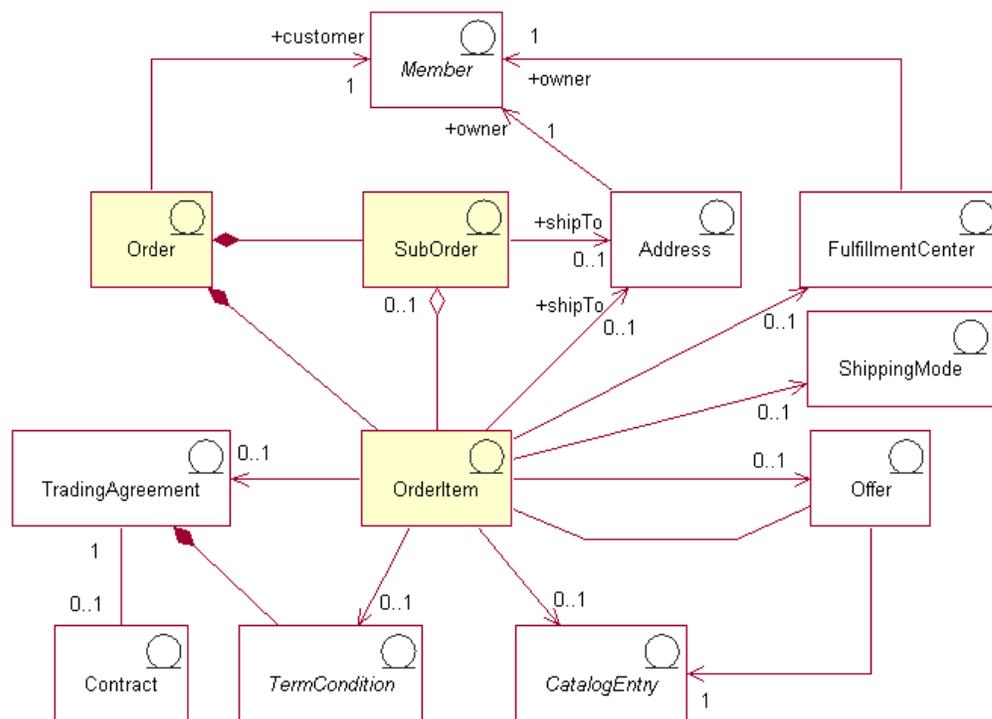
Una volta che un cliente ha selezionato una valuta di acquisto desiderata, tutti i pagamenti verranno eseguiti in quella valuta. In base al supporto e alle politiche di pagamento del negozio, i clienti possono pagare per i propri acquisti con pagamenti in linea (quando un cliente fornisce informazioni sul pagamento su Internet al sito del negozio) o con pagamenti non in linea (quando il cliente fornisce informazioni sul pagamento senza canali Internet, ad esempio per telefono o fax). Indipendentemente dai metodi di pagamento in linea o non in linea, i clienti che effettuano gli ordini devono fornire *informazioni sul pagamento*, comprese le informazioni di seguito riportate:

- Metodo di pagamento: il metodo di pagamento del cliente per l'ordine. In base alle cassette di pagamento configurate in Payment Manager per il negozio, è possibile impostare il negozio per accettare pagamenti non in linea, CyberCash per pagamenti in linea, le opzioni SET Secure Electronic Transaction e MIA (Merchant Initiated Authorization) per pagamenti in linea che non richiedono l'utilizzo da parte dei clienti di un portafoglio in linea o un metodo di pagamento personalizzato.
- Per i pagamenti con carta di credito, informazioni sulla carta: il tipo, il numero e la data di scadenza della carta di credito utilizzata dal cliente per pagare l'ordine. Le informazioni sulla carta di credito sono generalmente richieste se il negozio supporta il pagamento in linea.
- Numero di ordine di acquisto: il numero di ordine di acquisto che il cliente deve aver fornito nell'effettuare un ordine in un negozio. Il numero di ordine di acquisto autentica il cliente per effettuare un ordine dal negozio, come stipulato nei termini all'interno del contratto tra il negozio e il cliente.

Articoli ordine

Gli *articoli ordine* sono i singoli prodotti o articoli che costituiscono un ordine. Un ordine deve avere almeno un articolo. Ogni articolo dell'ordine rappresenta un oggetto che il cliente ha selezionato per l'acquisto. Inoltre, ogni articolo dell'ordine è associato a un accordo commerciale (generalmente un contratto), a una modalità di spedizione, a un centro di evasione ordini e ad una offerta speciale. Con ciascun articolo dell'ordine vengono memorizzati sconti, spese di spedizione e tasse.

Il grafico di seguito riportato illustra le risorse dell'articolo ordine di WebSphere Commerce. Dopo il grafico, è riportata la descrizione di ciascuna risorsa.



Questo grafico, insieme a tutti gli altri grafici della sezione relativa ai dati del negozio fa parte del modello di informazioni di WebSphere Commerce Server. Per ulteriori informazioni sul modello di informazioni, fare riferimento alla sezione "Modello di informazioni dati del negozio" a pagina 27. Per ulteriori informazioni sulle convenzioni utilizzate in questo grafico, fare riferimento all'Appendice A, "Legenda UML" a pagina 317.

Ordini secondari

Gli articoli dell'ordine sono raggruppati per formare *ordini secondari*. Un articolo secondario è la parte di un ordine che viene spedita ad un determinato indirizzo. Ad esempio, un cliente può indicare diversi indirizzi di spedizione per prodotti diversi contenuti nel carrello degli acquisti. Ciascun indirizzo di spedizione e i prodotti ad esso associati costituisce un ordine secondario. Gli articoli dell'ordine di un ordine secondario hanno lo stesso indirizzo di spedizione e possono essere utilizzati per visualizzare i subtotali degli importi degli articoli dell'ordine.

L'attributo quantità di un oggetto Articolo dell'ordine è un numero che può essere moltiplicato per l'attributo quantità nominale dell'oggetto Informazioni di spedizione sulle voci di catalogo associato all'oggetto Voce di catalogo per arrivare alla quantità effettiva rappresentata dall'articolo dell'ordine. L'oggetto Informazioni di spedizione sulle voci di catalogo specifica l'unità di misura in cui vengono indicate le quantità.

Anche se gli ordini vengono generalmente associati a un singolo negozio, esiste uno speciale tipo di ordine che può essere associato a un negozio o a un gruppo di negozi, denominato profilo dell'ordine. Questo profilo è rappresentato nel modello di oggetto come un ordine il cui stato è 'Q'. Il profilo dell'ordine viene utilizzato

per memorizzare le informazioni predefinite relative ad un cliente, ad esempio informazioni sul pagamento, la modalità di spedizione, l'indirizzo di spedizione e quello di fatturazione.

Altre risorse articolo dell'ordine

Un articolo dell'ordine può essere associato a nessuno o ad uno dei seguenti oggetti.

- Un indirizzo di spedizione relativo al cliente che ha effettuato l'ordine, contenente l'articolo dell'ordine. Durante l'elaborazione dell'ordine, il cliente deve specificare un indirizzo di spedizione, in modo che il centro di evasione ordini del negozio possa utilizzare questo indirizzo per spedire l'articolo dell'ordine correttamente.
- Un centro di evasione ordini per la spedizione e la ricezione degli articoli dell'ordine richiesti dai clienti e per la memorizzazione dell'inventario relativo a tali articoli.
- Una modalità di spedizione per l'articolo dell'ordine, vale a dire una combinazione di uno spedizioniere (una società che fornisce servizi di spedizione da un centro di evasione ordini ad un cliente) e di un servizio di spedizione offerto da quel corriere. Ad esempio, Società di spedizione ABC, servizio 24 ore su 24 e Società di spedizione ABC, consegna rapida sono modalità di spedizione.
- Un prezzo associato all'articolo dell'ordine. Includendo le offerte diverse nei diversi listini prezzi (o "contenitori posizioni commerciali"), i negozi possono presentare prezzi diversi per lo stesso prodotto o SKU a clienti diversi. Ad esempio, un'agenzia di viaggi può offrire biglietti aerei in quattro diversi listini prezzi: adulti, anziani, bambini e studenti.
- Una voce di catalogo relativa all'articolo dell'ordine, vale a dire che a ciascun articolo dell'ordine corrisponde una voce di catalogo.
- Un accordo commerciale che definisce i termini e le condizioni con cui l'articolo viene ordinato. Generalmente si tratta di un contratto, ma può trattarsi anche di una richiesta RFQ (Request for Quotation) vale a dire di una negoziazione, valida fino a quando l'ordine non viene inoltrato per l'elaborazione.



Per informazioni più dettagliate sulla struttura delle risorse ordine in WebSphere Commerce Server, fare riferimento ai modelli di oggetti e dati dell'ordine nella guida in linea di WebSphere Commerce.

Creazione di risorse ordine in WebSphere Commerce

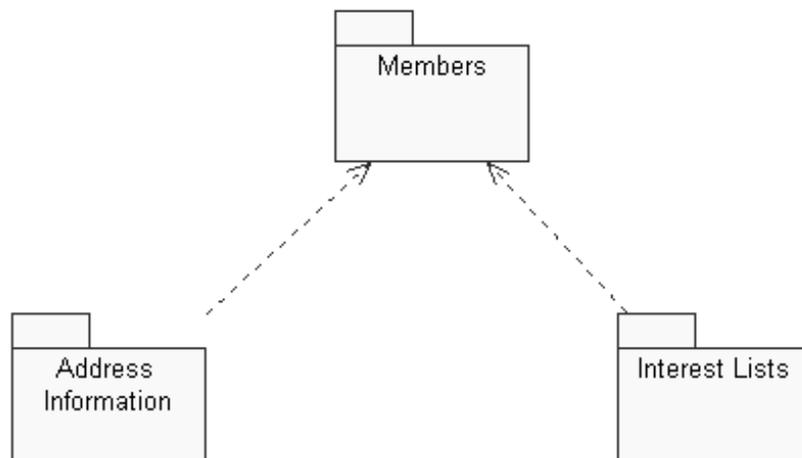
Un cliente può effettuare ordini da un negozio o richiedere l'assistenza di un rappresentante del servizio clienti per completare questa attività (mediante WebSphere Commerce Accelerator). Per creare un ordine per conto di un cliente B2C, fare riferimento alle sezioni "Creazione di un ordine per un cliente registrato" e "Creazione di un ordine per un cliente non registrato" della guida in linea di WebSphere Commerce. Per creare un ordine per conto di un cliente B2B, fare riferimento alla sezione della guida "Creazione di un ordine per un utente aziendale".

Capitolo 23. Risorse clienti e rivenditori

Un negozio può avere diversi utenti che partecipano alle attività del negozio, ma deve avere almeno un cliente e un rivenditore.

Risorse clienti in WebSphere Commerce

Un *cliente* è la persona che fa shopping al negozio, crea un elenco di interessi se lo desidera ed effettua ordini e acquisti dal negozio o dal rivenditore. Il grafico di seguito riportato illustra le risorse richieste da un cliente per effettuare un ordine da un negozio.



Questo grafico, insieme a tutti gli altri grafici della sezione relativa ai dati del negozio fa parte del modello di informazioni di WebSphere Commerce Server. Per ulteriori informazioni sul modello di informazioni, fare riferimento alla sezione "Modello di informazioni dati del negozio" a pagina 27. Per ulteriori informazioni sulle convenzioni utilizzate in questo grafico, fare riferimento all'Appendice A, "Legenda UML" a pagina 317.

Come illustrato nel grafico precedente, il sistema WebSphere Commerce contiene membri. A ciascun membro utente e a ciascun membro entità organizzativa può essere assegnato un ruolo.

Nota: In WebSphere Commerce, un *membro* può essere un'entità organizzativa, un utente o un gruppo di membri. Per ulteriori dettagli, consultare "Membri" a pagina 187.

In questo caso il membro utente è un cliente. Un cliente deve fornire informazioni sull'indirizzo e può avere un elenco di articoli di interesse. Il grafico illustra la relazione reciproca tra un membro (cliente) e le risorse cliente associate: per fare

shopping in un negozio, un cliente deve avere e fornire un indirizzo e può avere un elenco di interessi; l'indirizzo e l'elenco di interessi dipendono dall'esistenza di un cliente.

Informazioni sull'indirizzo

Quando effettua acquisti in un negozio, un cliente deve fornire tre tipi di informazioni sull'indirizzo: l'indirizzo di contatto, l'indirizzo di fatturazione e l'indirizzo di spedizione. Di seguito vengono descritti i tre tipi di indirizzo; ciascun indirizzo può essere univoco.

- Un indirizzo di contatto viene utilizzato per avvisare i clienti per vari scopi, quali lo stato o le modifiche ad un ordine o nuovi eventi relativi al negozio (ad esempio promozioni o manutenzione del negozio). L'indirizzo di contatto del cliente include il nome e il numero della strada, la città, lo stato o la provincia, il CAP o codice postale, il paese o la regione, l'indirizzo di e-mail, il numero di telefono e il numero di fax. Generalmente, l'indirizzo di contatto è l'indirizzo al quale il cliente può essere raggiunto più facilmente, ad esempio l'indirizzo dell'ufficio.
- Un indirizzo di fatturazione viene utilizzato per inviare un conto o una fattura. Un indirizzo di fatturazione include il nome e il numero della strada, la città, lo stato o la provincia, il CAP o codice postale, il paese o la regione, il numero di telefono e l'indirizzo di e-mail. L'indirizzo di fatturazione può corrispondere agli indirizzi di contatto o di spedizione.
- Un indirizzo di spedizione viene utilizzato per consegnare la merce acquistata. Un indirizzo di spedizione include il nome e il numero della strada, la città, lo stato o la provincia, il CAP o codice postale, il paese o la regione, il numero di telefono e l'indirizzo di e-mail. L'indirizzo di spedizione può corrispondere agli indirizzi di contatto o di fatturazione.

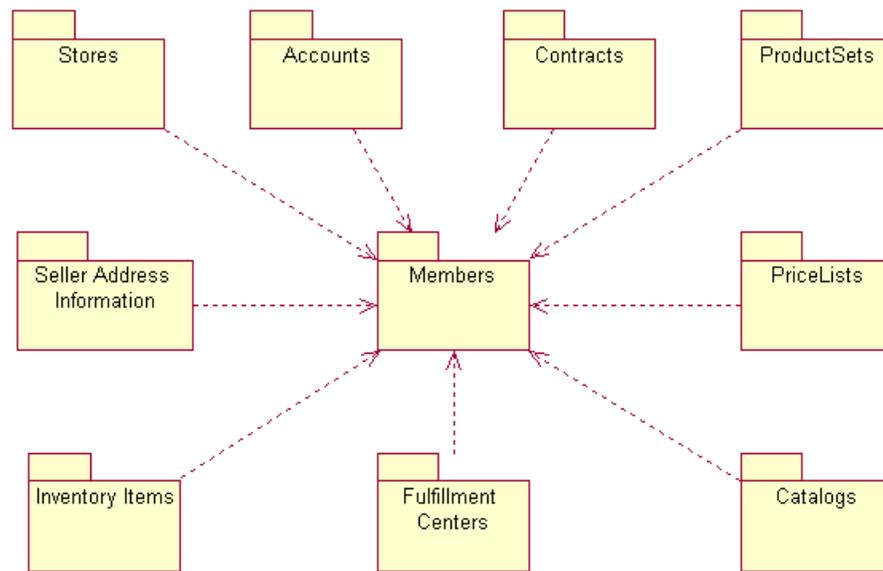
Elenchi di interessi

I negozi possono supportare *elenchi di interessi*. Ciò significa che i clienti aggiungono ai propri elenchi di interessi i prodotti che vorrebbero ordinare in futuro. Un elenco di interessi non è un carrello degli acquisti, può contenere articoli di più negozi ma non contiene prezzi, indirizzi di spedizione, modalità di spedizione, informazioni sulla disponibilità degli inventari o importi calcolati, quali sconti, tasse di spedizione e tasse.

Comprensione delle risorse del venditore in WebSphere Commerce

Un *rivenditore* supervisiona la gestione e gli obiettivi generali del negozio e tiene traccia delle vendite del negozio. Un rivenditore vende merci e servizi al cliente. Il ruolo del rivenditore è equivalente a quello di un comune rivenditore e ha accesso a tutte le funzioni di WebSphere Commerce Accelerator. Il grafico di seguito riportato illustra le risorse richieste da un rivenditore per la gestione di un negozio

e la vendita ai clienti.



Questo grafico, insieme a tutti gli altri grafici della sezione relativa ai dati del negozio fa parte del modello di informazioni di WebSphere Commerce Server. Per ulteriori informazioni sul modello di informazioni, fare riferimento alla sezione "Modello di informazioni dati del negozio" a pagina 27. Per ulteriori informazioni sulle convenzioni utilizzate in questo grafico, fare riferimento all'Appendice A, "Legenda UML" a pagina 317.

Come illustrato nel grafico precedente, il sistema WebSphere Commerce contiene membri. A ciascun membro viene assegnato un ruolo, ad esempio responsabile del servizio clienti per il negozio o responsabile dell'inventario nel magazzino. Un rivenditore può gestire le seguenti risorse per la vendita ai clienti:

- Negozi
- Conti (facoltativi)
- Contratti (o almeno il contratto predefinito di WebSphere Commerce)
- Gruppi di prodotti
- Listino prezzi
- Cataloghi
- Centri di evasione ordini
- Articoli di inventario

Il grafico precedente illustra la relazione tra un membro (rivenditore) e le risorse rivenditore associate; ciò significa che un rivenditore può avere le risorse sopra elencate per la gestione di un negozio e le risorse necessarie per la distribuzione.

Negozi

Un *negozio* in linea di WebSphere Commerce è costituito da un insieme di file HTML e JSP e da risorse tasse, spedizione, pagamento, catalogo e altre risorse database contenute in un archivio negozio. Un negozio contiene inoltre i dati del negozio, vale a dire informazioni presenti nel database di WebSphere Commerce che consentono il funzionamento del negozio.

Per ulteriori informazioni sui negozi di WebSphere Commerce, fare riferimento al Capitolo 6, "Risorse del negozio" a pagina 43 e alla Parte 4, "Sviluppo dei dati del negozio" a pagina 37.

Conti

Un negozio può impostare *conti* commerciali per i clienti per consentire loro di fare shopping nel negozio. Un conto contiene le seguenti informazioni:

- Il nome del conto, che spesso corrisponde al nome dell'organizzazione alla quale il cliente è associato. Tale organizzazione ha definito i contratti con il negozio, stipulando i termini per gli acquisti del cliente nel negozio. Ad esempio, l'organizzazione IBM potrebbe avere dei contratti con una società di forniture per ufficio, ABC Office Supplies Company.
- Il nome del rappresentante, vale a dire il nome dell'organizzazione del rappresentante all'interno dell'organizzazione del venditore responsabile del conto.
- Il numero di contratti che appartengono al conto.

Per ulteriori informazioni sui conti WebSphere Commerce, fare riferimento alla sezione "Conto (conto commerciale)" a pagina 98 e alla guida in linea di WebSphere Commerce.

Contratti

Generalmente, in WebSphere Commerce, tutti i clienti devono fare shopping sotto *contratto*. Ogni conto tra il cliente e il rivenditore deve essere associato a uno o più contratti (o almeno ad un *contratto predefinito* per i clienti non registrati o per i clienti che acquistano prodotti non coperti da altri contratti). Un contratto consente al cliente di acquistare i prodotti di un negozio ad un prezzo specificato, per un determinato periodo di tempo, in base ai termini, alle condizioni e alle politiche aziendali stipulate nel contratto. Il rivenditore distribuisce il contratto in modo che i clienti possano fare shopping nel negozio.

Per ulteriori informazioni sui contratti di WebSphere Commerce ed il contratto predefinito che un rivenditore può utilizzare, fare riferimento alla sezione "Contratto" a pagina 99.

Gruppi di prodotti

I *gruppi di prodotti* forniscono un meccanismo per il quale un rivenditore organizza i cataloghi in linea in sottoinsiemi logici, in modo da consentire ai vari clienti di trarre vantaggio dalle diverse visualizzazioni del catalogo. Un rivenditore, inoltre, può creare un contratto per un cliente e stipulare che il cliente acquisti i prodotti di un insieme predefinito.

Per ulteriori informazioni sugli insiemi di prodotti di WebSphere Commerce, fare riferimento alla sezione "Gruppi di prodotti" a pagina 58.

Listino prezzi

Un *listino prezzi* è associato al prezzo che un rivenditore offre o presenta ad un cliente. Un rivenditore può presentare prezzi diversi per lo stesso prodotto a clienti diversi. In WebSphere Commerce, un'offerta è conosciuta anche come *posizione commerciale* e rappresenta il prezzo di una voce di catalogo e i criteri che il cliente deve soddisfare per qualificarsi per quel prezzo.

In WebSphere Commerce, un oggetto Offerta è parte di un contenitore di posizioni commerciali, che è di proprietà di un membro. Un contenitore di posizioni commerciali contiene posizioni commerciali e può essere disponibile per tutti i clienti o solo per i clienti di determinati gruppi, mediante accordi commerciali o contratti. A volte un contenitore di posizioni commerciali è un listino prezzi. Esistono due tipi di listino prezzi: un listino prezzi standard contenente i prezzi di base per i prodotti del catalogo del negozio e un listino prezzi personalizzato, che specifica l'elenco dei prodotti ed i relativi prezzi personalizzati.

Per ulteriori informazioni sui listini prezzi di WebSphere Commerce, fare riferimento al Capitolo 9, "Risorse assegnazione prezzi" a pagina 89.

Cataloghi

Un negozio di WebSphere Commerce utilizza almeno un *catalogo* in linea per la visualizzazione delle merci e dei servizi che il rivenditore offre per la vendita. Generalmente, un catalogo in linea contiene prezzi, immagini e descrizioni degli articoli in vendita. Un catalogo in linea può presentare la merce in diverse categorie per facilitare la navigazione.

Ciascun negozio nel sistema WebSphere Commerce deve avere un *catalogo principale*, utilizzato per la gestione dei cataloghi. E' il singolo catalogo che contiene tutti i prodotti, articoli, relazioni e prezzi standard per tutto ciò che è in vendita nel negozio. Se un rivenditore ha più di un negozio, il catalogo principale può essere condiviso tra i vari negozi.

Per ulteriori informazioni sugli insiemi di prodotti di WebSphere Commerce, fare riferimento al Capitolo 8, "Risorse catalogo" a pagina 55.

Centri di evasione ordini

I *centri di evasione ordini* vengono utilizzati dai negozi sia come magazzini di inventario che come centri di spedizione e ricezione. Ad un rivenditore può essere associato uno o più centri di evasione ordini.

Da una prospettiva del server di WebSphere Commerce, un oggetto centro di evasione ordini è separato dall'oggetto Negozio. Esso gestisce la spedizione e l'inventario dei prodotti. Per inviare un ordine, il centro di evasione ordini si basa su un oggetto Modalità di spedizione specificato dal cliente. L'oggetto Modalità di spedizione indica lo spedizioniere ed il metodo di spedizione da utilizzare per l'evasione ordini. In un centro di evasione ordini, l'oggetto Disposizione di spedizione indica che un negozio è stato organizzato con un oggetto Centro di evasione ordini per spedire i prodotti utilizzando una certa modalità di spedizione.

Per ulteriori informazioni sui centri di evasione ordini di WebSphere Commerce, fare riferimento al Capitolo 11, "Risorse evasione ordini" a pagina 109 e al Capitolo 18, "Risorse spedizione" a pagina 135.

Articoli di inventario

Gli *articoli di inventario* includono qualsiasi cosa possa essere considerata fisicamente in un centro di evasione ordini. Il sistema WebSphere Commerce definisce tipi specifici di inventario, quali articoli, prodotti, SKU, bundle e pacchetti; ma tali elementi sono tutti considerati inventario. I prodotti sono configurati per l'evasione ordini mediante gli strumenti di Product Manager su WebSphere Commerce Accelerator.

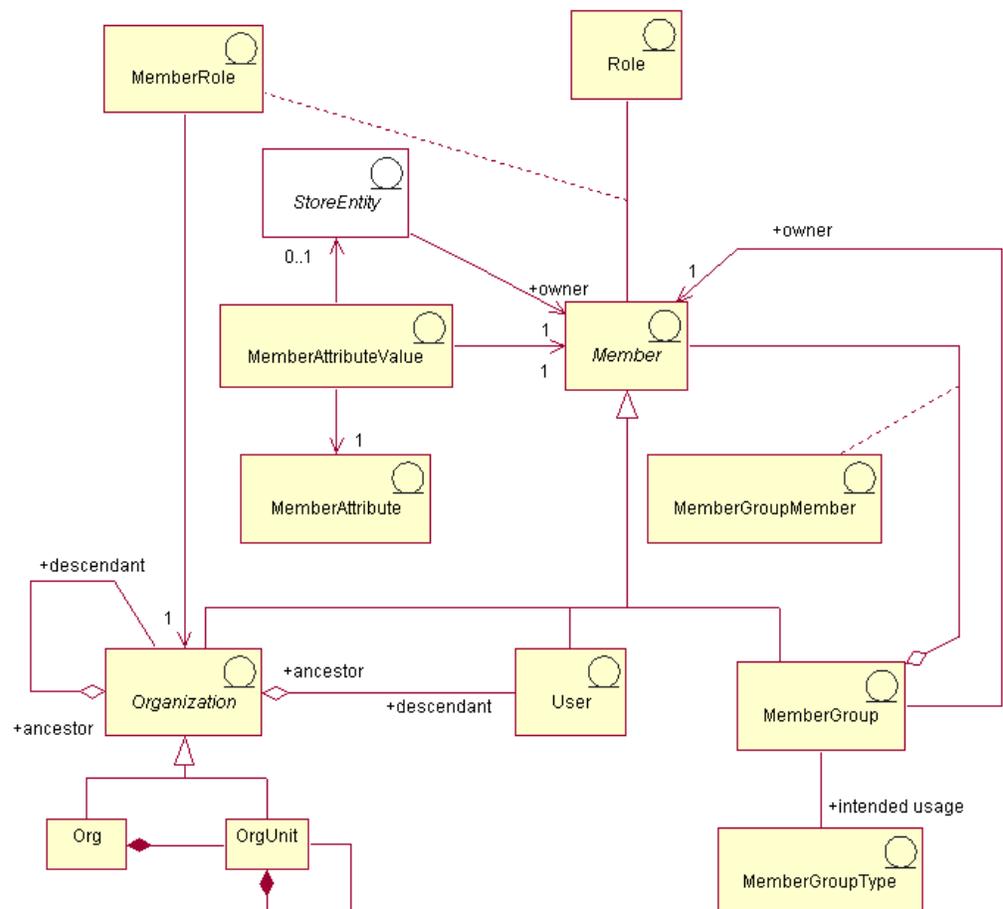
Per ulteriori informazioni sugli articoli di inventario di WebSphere Commerce, fare riferimento alla guida in linea di WebSphere Commerce e al Capitolo 21, "Risorse inventario" a pagina 173.

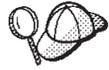
Risorse membri in WebSphere Commerce

Le risorse membri di WebSphere Commerce includono i dati per i partecipanti del sistema WebSphere Commerce. Un membro può essere un utente, un gruppo di utenti o un'entità organizzativa. L'assegnazione ruoli a utenti e entità organizzative spetta a un responsabile, come il Responsabile del sito. Una volta che viene assegnato un ruolo ad un membro, il componente del controllo degli accessi autorizza il membro a partecipare alle attività associate al ruolo. Ad esempio, un'organizzazione può essere un acquirente, un venditore o entrambi. E' possibile assegnare più ruoli all'utente. Un responsabile può creare gruppi di membri, vale a dire gruppi di utenti classificati per vari motivi aziendali. Utilizzare la console di gestione di WebSphere Commerce per creare e gestire organizzazioni, utenti, ruoli e gruppi di membri.

La logica aziendale per le risorse membri fornisce servizi di registrazione membri e di gestione del profilo. Gli altri servizi che sono strettamente collegati alle risorse membri comprendono il controllo accessi, l'autenticazione e la gestione sessione. Per ulteriori dettagli su questi argomenti, fare riferimento alla guida in linea di WebSphere Commerce.

Il grafico di seguito riportato illustra le risorse membri di WebSphere Commerce. Dopo il grafico, è riportata la descrizione di ciascuna risorsa.





Questo grafico, insieme a tutti gli altri grafici della sezione relativa ai dati del negozio fa parte del modello di informazioni di WebSphere Commerce Server. Per ulteriori informazioni sul modello di informazioni, fare riferimento alla sezione "Modello di informazioni dati del negozio" a pagina 27. Per ulteriori informazioni sulle convenzioni utilizzate in questo grafico, fare riferimento all'Appendice A, "Legenda UML" a pagina 317.

Membri

Un *membro* in WebSphere Commerce può essere uno dei seguenti elementi:

- Un'entità *organizzativa*. L'entità organizzativa può essere un'organizzazione, come la "IBM" o un'unità organizzativa di un'organizzazione come la "Divisione commercio elettronico" dell'IBM.
- Un *utente* (registrato o non registrato). Un utente registrato dispone di un identificativo univoco, di una password e gli viene richiesto di fornire alcuni dati del profilo per la registrazione. Gli utenti registrati possono essere classificati secondo il tipo di profilo: il tipo 'B' denota un utente commerciale (o un cliente B2B) e il tipo C denota un utente al dettaglio (o un cliente B2C). Per ulteriori informazioni sugli utenti registrati e non registrati, fare riferimento alla sezione "Membri" della guida in linea di WebSphere Commerce.
- Un *gruppo di membri*. Si tratta di un gruppo di utenti catalogati per vari motivi aziendali. E' possibile utilizzare i raggruppamenti per il controllo degli accessi, per l'approvazione e per scopi di marketing, come il calcolo di sconti, prezzi e la visualizzazione dei prodotti.

Ogni *entità negozio* (vale a dire negozio o gruppo di negozi) è di proprietà di un membro.

Attributi membro

Un membro di WebSphere Commerce presenta un insieme di *attributi* ed a ciascun attributo è associato un *valore*. Un profilo utente di base per il membro include informazioni demografiche, sulla registrazione, sull'indirizzo, sulla cronologia degli acquisti e su altri tipi di attributi.

Un profilo utente aziendale contiene le stesse informazioni di un profilo utente di base e informazioni sul lavoro, quali il numero dell'impiegato, la qualifica o una descrizione dell'attività svolta. Durante la registrazione, gli utenti aziendali devono identificare l'organizzazione a cui appartengono. I profili per l'organizzazione includono queste informazioni aggiuntive, come il nome organizzazione e la categoria commerciale.

Le regole relative al controllo degli accessi definiscono l'utente che può eseguire la gestione dei profili. I profili membri possono contenere una varietà di attributi personali collegati al commercio (come ruoli, informazioni sul pagamento, indirizzi, lingue e valute preferite e dispositivi di elaborazione invadente). Gli attributi possono essere relativi al negozio di appartenenza. Questi attributi non sono supportati per i gruppi di membri.

Ruoli

Ogni membro può svolgere uno o più ruoli in un'organizzazione. Un Responsabile del sito assegna un ruolo o più ruoli a ciascun utente membro. Ad esempio, come membro dell'organizzazione IBM il ruolo di John Smith come responsabile del servizio clienti indica che John esegue attività per i clienti dell'IBM e li assiste nelle questioni relative alla registrazione, agli ordini o alle restituzioni. John può avere anche il ruolo di supervisore del servizio clienti, che ha tutte le responsabilità delle attività descritte sopra oltre all'autorità per approvare e supervisionare le attività di tutti gli altri responsabili del servizio clienti.

Il sistema WebSphere Commerce fornisce il seguente insieme di tipi di ruoli predefiniti:

- Ruoli per lo sviluppo del sito e del contenuto
- Ruoli delle operazioni tecniche
- Ruoli di gestione del marketing
- Ruoli per la gestione dei prodotti
- Ruoli per la gestione delle relazioni aziendali
- Ruoli per la gestione logistica e delle operazioni
- Ruoli per la gestione organizzativa

Per i dettagli su ciascuno di questi ruoli, fare riferimento alla sezioni Ruoli della guida in linea di WebSphere Commerce. Un Responsabile del sito può assegnare questi ruoli oltre ai nuovi ruoli creati dal Responsabile del sito, in base all'entità organizzativa; ciò significa che gli utenti che appartengono ad un'entità organizzativa possono assumere i ruoli assegnati a quell'entità organizzativa.

Quando viene assegnato un ruolo a un utente, tale ruolo ha lo scopo di entità organizzativa. Quando ad un utente è assegnato un ruolo, l'utente non svolge necessariamente quel ruolo per l'entità organizzativa a cui appartiene; ciò significa che, quando un responsabile esegue l'assegnazione, tale responsabile può selezionare l'entità organizzativa per cui l'utente svolge quel ruolo. Se il responsabile seleziona l'Organizzazione principale, l'utente svolge quel ruolo per tutte le entità organizzative.



Per informazioni più dettagliate sulla struttura delle risorse membri di WebSphere Commerce, fare riferimento ai modelli di oggetti e dati nella guida in linea di WebSphere Commerce.

Creazione di risorse membri in WebSphere Commerce

Per creare un rivenditore (un'organizzazione che agisca come proprietario del negozio) e gestire le informazioni sul rivenditore, utilizzare la console di gestione. Per ulteriori informazioni, fare riferimento alla sezione "Creazione di un'organizzazione" nella guida in linea di WebSphere Commerce.

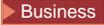
Un cliente non viene creato dallo sviluppatore del negozio; quando un cliente effettua la registrazione ad un negozio, le informazioni sulla registrazione vengono raccolte e gestite dal sistema WebSphere Commerce.

Parte 5. Aggiunta del controllo accessi al negozio

Capitolo 24. Controllo dell'accesso al negozio

WebSphere Commerce consente di stabilire, mediante il controllo accessi, le attività che un particolare utente (cliente o responsabile) può eseguire. Questo capitolo illustra come aggiungere il controllo accessi al negozio, limitando così le pagine che un utente può visualizzare e le attività che può eseguire.

Per ulteriori informazioni sul modello di controllo accessi in WebSphere Commerce, e sull'applicazione del controllo accessi a livello del sito, fare riferimento al manuale *IBM WebSphere Commerce Access Control Guide*. Per ulteriori informazioni sull'implementazione del controllo accessi nel codice personalizzato, fare riferimento al manuale *IBM WebSphere Commerce - Guida per il programmatore*. Questi manuali sono disponibili ai seguenti URL:

 http://www.ibm.com/software/webservers/commerce/wc_be/lit-tech-general.html

 http://www.ibm.com/software/webservers/commerce/wc_pe/lit-tech-general.html

Comprensione del controllo accessi in WebSphere Commerce

Il modello di controllo accessi per WebSphere è descritto nei dettagli nel manuale *IBM WebSphere Commerce Access Control Guide*. Viene qui fornita una breve panoramica che consente di comprendere l'importanza di questa funzione per lo sviluppo di un negozio.

Il controllo accessi in WebSphere Commerce è composto dai seguenti elementi: utenti, azioni, risorse e relazioni.

- Gli utenti sono le persone che utilizzano il sistema. Per il controllo, è necessario suddividere gli utenti in gruppi di accesso. Ad esempio, suddividere gli utenti del negozio in gruppi di accesso quali clienti registrati e clienti ospiti, o in gruppi di gestione, ad esempio rappresentati dell'assistenza clienti.
- Le azioni sono attività che gli utenti possono eseguire sulla risorsa. Per il controllo, è necessario suddividere anche le azioni in gruppi di azioni. Ad esempio, un'azione comune utilizzata in un negozio è una vista. Una vista viene richiamata per visualizzare una pagina del negozio ai clienti. Le viste utilizzate nel negozio devono essere assegnate a un gruppo di azioni.
- Le risorse sono entità protette. Ad esempio, se l'azione è una vista, la relativa risorsa è la classe `com.ibm.commerce.command.ViewCommand`. Per il controllo, le risorse sono suddivise in gruppi di risorse.
- Le relazioni sono i rapporti che intercorrono tra l'utente e la risorsa. Le politiche di controllo accessi possono richiedere che venga soddisfatta una relazione tra l'utente e la risorsa.

Politiche di controllo accessi

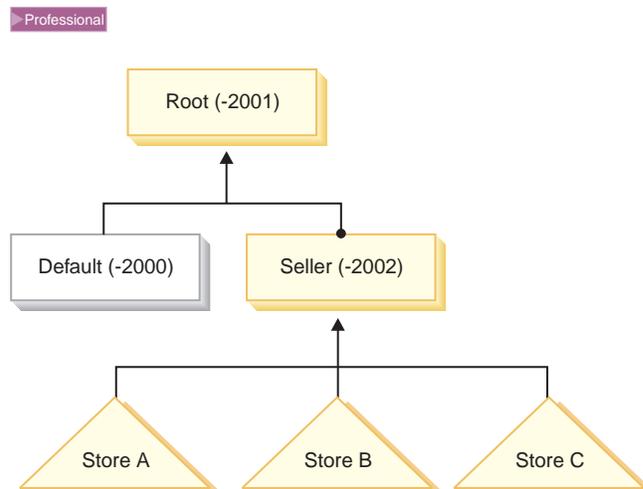
Le politiche di controllo accessi autorizzano i gruppi di accesso ad eseguire particolari azioni su una risorsa di WebSphere Commerce se gli utenti appartenenti al gruppo soddisfano una determinata relazione con la suddetta risorsa.

WebSphere Commerce fornisce più di duecento politiche di controllo accessi predefinite che vengono caricate durante la creazione dell'istanza. Tali politiche coprono un'ampia gamma di comuni attività commerciali, incluso l'elaborazione e la creazione degli ordini e la negoziazione, ad esempio **Business** RFQ (request for quotes) e **Business** contratti. Le politiche predefinite sono descritte nel manuale *IBM WebSphere Commerce Access Control Guide*

Ogni politica di controllo accessi appartiene a un'entità organizzativa. Una politica di controllo accessi può essere applicata solo alle risorse del proprietario della politica.

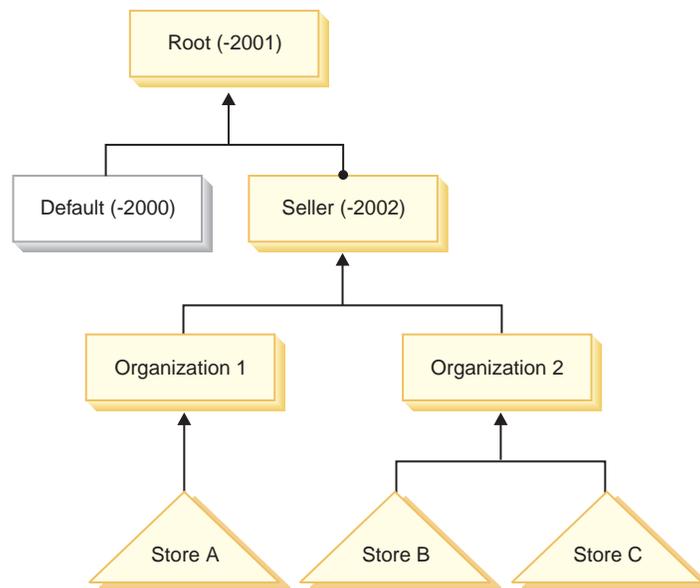
Nota: In termini di controllo accessi, la proprietà delle risorse ha uno speciale significato. Tutte le risorse devono implementare l'interfaccia `com.ibm.commerce.security.Protectable`. Uno dei metodi che è possibile eseguire su tale interfaccia è `getOwner()`, che restituisce l'ID membro del proprietario della risorsa. Ad esempio, il bean entità Ordine è una risorsa protetta perché la relativa interfaccia remota estende l'interfaccia protetta. L'implementazione di `getOwner()` del suddetto bean fa in modo che una specifica risorsa Ordine restituisca il proprietario del negozio in cui è stato eseguito l'ordine. Per le politiche in cui la risorsa è un comando, ad esempio, `com.ibm.commerce.command.ViewCommand`, l'implementazione predefinita di `getOwner()` restituisce il proprietario del negozio corrente nel contesto del comando. Se nel contesto del comando non è presente alcun negozio, viene utilizzata come proprietario l'Organizzazione principale. Per ulteriori informazioni, consultare il manuale *IBM WebSphere Commerce - Guida per il programmatore*.

Considerare i seguenti grafici:



In WebSphere Commerce Professional Edition l'Organizzazione principale è l'organizzazione con il livello più alto. Essa possiede tutte le entità organizzative. Di conseguenza, le politiche di controllo accessi che appartengono ad essa si applicano a tutte le risorse del sito. Se una politica di controllo accessi appartiene all'Organizzazione venditrice si applica solo alle risorse del venditore.

► Business



In WebSphere Commerce Business Edition l'Organizzazione principale è l'organizzazione con il livello più alto. Essa possiede tutte le entità organizzative. Di conseguenza, le politiche di controllo accessi che appartengono ad essa si applicano a tutte le risorse del sito. Se una politica di controllo accessi appartiene all'Organizzazione venditrice si applica solo alle risorse del venditore, cioè solo alle risorse dell'Organizzazione 1 e 2, inclusi i Negozi A, B e C. Se appartiene all'Organizzazione 1, si applica solo alle risorse comprese nel Negozio A. Se appartiene all'Organizzazione 2, si applica solo alle risorse dei Negozi B e C.

► Business

Poiché le politiche di controllo accessi appartengono ad entità organizzative, se dopo aver creato più negozi nel sito si desidera applicare politiche di controllo diverse ai singoli negozi, è necessario creare un'organizzazione distinta per ciascun negozio.

Nota: La maggior parte delle politiche di controllo accessi fornite con WebSphere Commerce appartengono all'Organizzazione principale e si applicano a tutte le risorse del sito. Le politiche di controllo accessi che appartengono all'Organizzazione principale sono dette politiche a livello del sito. Le politiche che appartengono ad entità organizzative sono dette politiche a livello dell'organizzazione.

Il controllo accessi nei negozi

Tutti i negozi creati in WebSphere Commerce sono soggetti alle politiche di controllo accessi predefinite che appartengono dell'Organizzazione principale. Le risorse presenti nei negozi sono soggette anche alle politiche di controllo accessi dell'organizzazione a cui appartiene il negozio e a qualsiasi politica dei prototipi di tale organizzazione.

Per impostazione predefinita, l'Organizzazione principale possiede la maggior parte delle politiche di controllo accessi. Tuttavia, quando si crea un negozio basato sui negozi di esempio forniti con WebSphere Commerce, vengono create nuove politiche di controllo accessi possedute dall'organizzazione a cui appartiene il

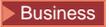
negozio. Per informazioni sulle politiche di controllo accessi nei negozi di esempio, fare riferimento a “Controllo accessi nei negozi di esempio”.

Quando si crea un negozio, che sia o meno basato su un esempio, è possibile creare politiche di controllo accessi o modificare quelle esistenti. Le nuove politiche si applicheranno solo ai negozi appartenenti a quella organizzazione. Ad esempio, se si creano nuove viste per visualizzare le pagine del negozio, è necessario assegnare le politiche di controllo accessi a tali viste.

I dati sul controllo accessi a livello dell'organizzazione sono definiti nei file delle politiche di controllo accessi di alto livello. Tali file definiscono le possibili azioni, i gruppi di azioni, le risorse, i gruppi di risorse e le relazioni che possono essere utilizzate da una politica. Definiscono, inoltre, le politiche specifiche per una particolare organizzazione. I negozi di esempio forniti con WebSphere Commerce contengono file delle politiche di controllo accessi di alto livello. La sezione seguente illustra il modo in cui i negozi di esempio utilizzano questi file per definire le informazioni sull'organizzazione.

Controllo accessi nei negozi di esempio

Tutti i negozi di esempio contengono file delle politiche di controllo accessi di alto livello che definiscono le politiche di controllo accessi create specificamente per il negozio. Tali politiche appartengono all'organizzazione che possiede il negozio. Vengono di seguito descritte le politiche di controllo accessi definite per i negozi di esempio NewFashion e  ToolTech.

Le politiche di controllo accessi create per  ToolTech sono le seguenti:

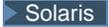
- AllUsersForToolTechExecuteToolTechAllUsersViews
- RegisteredApprovedUsersForToolTechExecuteToolTechRegisteredApprovedUsersViews

La politica di controllo accessi creata per NewFashion è la seguente:

- AllUsersExecuteNewFashionAllUsersViews

Tali politiche determinano quali utenti possono visualizzare le viste dei negozi di esempio o dei negozi basati sugli esempi.

Le informazioni sul controllo accessi sono definite in due file delle politiche di controllo accessi di alto livello che definiscono le possibili azioni, i gruppi di azioni, le risorse, i gruppi di risorse e le definizioni di politiche utilizzate nei negozi di esempio: *nomenegozioesempioAccessPolicies.xml* e *enomenegozioesempioAccessPolicies_locale.xml*. Questi file si trovano nella seguente directory:

-  unità:\WebSphere\CommerceServer\samples\stores\nomenegozioesempio
-  unità:\ProgramFiles\WebSphere\CommerceServer\samples\stores\nomenegozioesempio
-  /usr/WebSphere/CommerceServer/samples/stores/nomenegozioesempio
-  /opt/WebSphere/CommerceServer/samples/stores/nomenegozioesempio
-  /opt/WebSphere/CommerceServer/samples/stores/nomenegozioesempio
-  /QIBM/ProdData/WebCommerce/samples/stores/nomenegozioesempio

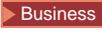
dove *nomenegozioesempio* è il nome dell'archivio negozio di esempio su cui si basa il negozio, ad esempio NewFashion.

Prima di impacchettare i negozi di esempio come archivio negozio, questi due file delle politiche di controllo accessi di alto livello vengono convertiti in file XML che possono essere utilizzati con il pacchetto Loader. Questi file XML convertiti vengono poi impacchettati nell'archivio negozio e pubblicati con il resto dell'archivio.

Comprensione dei file delle politiche di controllo accessi dei negozi di

esempio: Per capire come il controllo viene aggiunto a livello del negozio, è necessario acquisire familiarità con i file delle politiche di controllo accessi dei negozi di esempio di alto livello. Gli esempi di seguito riportati sono stati ricavati dal file  ToolTechAccessPolicies.xml.

Definizione delle azioni: La prima sezione del file

 ToolTechAccessPolicies.xml definisce le nuove azioni del negozio non coperte da politiche di controllo accessi esistenti. In questo caso, le azioni sono tutte le viste utilizzate nel negozio. Per visualizzare una pagina del negozio tramite una vista che può essere richiamata direttamente da un URL o avviata mediante reindirizzamento da un altro comando (invece di essere avviata inoltrandola alla vista), è necessario definirla come azione. Considerare il seguente esempio:

```
<!-- [Start of Action definitions] -->
<!-- [this is the dictionary of possible actions -->
<Action Name="GenericApplicationError"
      CommandName="GenericApplicationError">
  </Action>

<Action Name="GenericSystemError"
      CommandName="GenericSystemError">
  </Action>

<Action Name="OrderOptionsView"
      CommandName="OrderOptionsView">
  </Action>

<!--[End of Action definitions] -->
```

dove

- Action Name è l'etichetta utilizzata per fare riferimento a quest'azione nel file XML. In questi esempi, l'etichetta corrisponde al nome della vista.
- CommandName è il nome della vista memorizzata nella colonna VIEWNAME della tabella VIEWREG. CommandName sarà memorizzato nella colonna Action della tabella AACTION.

Nota: Se un'azione è già definita nelle politiche predefinite oppure in un'altra politica di WebSphere Commerce, non è necessario ridefinirla per ciascuna politica da cui viene utilizzata.

Definizione delle categorie delle risorse: La seconda sezione del file definisce le categorie delle risorse. Una categoria delle risorse si riferisce a una classe di risorse. Le categorie delle risorse identificate per ToolTech sono le classi `com.ibm.commerce.command.ViewCommand` e `com.ibm.commerce.tools.command.ToolsForwardViewCommand`.

```
<!-- [Start of Resource Category definitions] -->
<!-- the dictionary of Protectable resources -->
```

```

<ResourceCategory Name="com.ibm.commerce.command.ViewCommandResourceCategory"
  ResourceBeanClass="com.ibm.commerce.command.ViewCommand">

</ResourceCategory>

<ResourceCategory Name="com.ibm.commerce.tools.command.
ToolsForwardViewCommandResourceCategory"
  ResourceBeanClass="com.ibm.commerce.tools.command.ToolsForwardViewCommand">

</ResourceCategory>
<!-- [End of Resource Category definitions] -->

```

dove

- ResourceCategory Name è l’etichetta utilizzata per fare riferimento a questa categoria di risorse nel file XML.
- ResourceBeanClass è il nome della classe.

Nota: Se una categoria delle risorse è già definita nelle politiche predefinite o in un’altra politica di WebSphere Commerce, non è necessario ridefinirla per ciascuna politica da cui viene utilizzata. Nell’esempio sopra riportato la categoria delle risorse è già definita nelle politiche predefinite, ma è nuovamente definita a scopi illustrativi.

Definizione dei gruppi di azioni: La terza sezione definisce il gruppo di azioni. Il gruppo di azioni è un raggruppamento di azioni definite nella prima sezione del file. Nell’esempio ToolTech, tutte le nuove viste utente vengono riunite nel gruppo ToolTechAllUserViews, utilizzato in una politica che consente a tutti utenti di accedere alle viste, o nel gruppo ToolTechRegisteredApprovedUserViews, utilizzato in una politica che consente solo agli utenti registrati di accedere alle viste.

Nota: E’ anche possibile aggiungere ai gruppi azioni definite altrove in WebSphere Commerce. Se definite altrove in WebSphere Commerce, le azioni non devono essere definite nell’elenco Accesso discusso in “Definizione delle azioni” a pagina 195.

```

<!-- [Start of Action Group definitions] -->
<!-- Dictionary of grouped actions usable in policies -->
<!-- cross-component view-related action groups -->
<ActionGroup Name="ToolTechAllUsersViews"
  OwnerID="RootOrganization">

  <ActionGroupAction Name="UserRegistrationForm"/>
  <ActionGroupAction Name="UserRegistrationErrorView"/>
  <ActionGroupAction Name="GenericApplicationError"/>
  <ActionGroupAction Name="GenericSystemError"/>
  <ActionGroupAction Name="LogonForm"/>
</ActionGroup>

<!-- [End of Action Group definitions] -->

```

dove

- ActionGroup Name è il nome del gruppo di azioni.
- OwnerID è il proprietario del gruppo di azioni. L’Organizzazione principale deve essere il proprietario del gruppo di azioni.
- ActionGroupAction Name è il nome di un’azione che appartiene al gruppo. ActionGroupAction Name deve corrispondere al nome definito nell’elemento Action Name in “Definizione delle azioni” a pagina 195.

Definizione dei gruppi di risorse: La sezione successiva definisce i gruppi di risorse. Un gruppo di risorse identifica un insieme di risorse correlate.

```

<!-- [Start of Resource Group definitions] -->
<!-- Dictionary of grouped resources usable in policies -->

<!-- Grouped resources permitting view execution,
by any command extending com.ibm.commerce.command.ViewCommand
(with or without the Tools Framework) -->
<ResourceGroup Name="ViewCommandResourceGroup" OwnerID="RootOrganization">

  <ResourceGroupResource Name="com.ibm.commerce.command.ViewCommandResourceCategory"/>
</ResourceGroup>

</ResourceGroup> !-- [End of Resource Group definitions] -->

```

dove

- ResourceGroup Name è il nome del gruppo di risorse.
- OwnerID è il proprietario del gruppo di risorse. L'Organizzazione principale deve essere il proprietario del gruppo di risorse.
- ResourceGroupResource Name è il nome della categoria di risorse inclusa nel gruppo.

Definizione delle politiche: La sezione finale definisce le nuove politiche utilizzate nel negozio.

```

!-- [Start of Policy definitions] -->

!-- AllUsers for ToolTech can execute ToolTechAllUsersViews -->

<Policy Name="AllUsersForToolTechExecuteToolTechAllUsersViews"
  OwnerID="MEMBER_ID"
  UserGroup="AllUsers"
  UserGroupOwner="RootOrganization"
  ActionGroupName="ToolTechAllUsersViews"
  ResourceGroupName="ViewCommandResourceGroup">
  </Policy>
!-- RegisteredApprovedUsers for ToolTech can execute
ToolTechRegisteredApprovedUsersViews -->

<Policy Name="RegisteredApprovedUsersFor
ToolTechExecuteToolTechRegisteredApprovedUsersViews"
  OwnerID="MEMBER_ID"
  UserGroup="RegisteredApprovedUsers"
  UserGroupOwner="RootOrganization"
  ActionGroupName="ToolTechRegisteredApprovedUsersViews"
  ResourceGroupName="ViewCommandResourceGroup">
  </Policy>

!-- [End of of Policy definitions] -->

```

dove

- Policy Name è il nome della politica da definire.
- OwnerId è il proprietario della politica. In questo caso il proprietario della politica è l'organizzazione che possiede il negozio.
- UserGroup è il gruppo di utenti (il gruppo di accesso) al quale si applica la politica.
- UserGroupOwner è il proprietario del gruppo di accesso. In questo esempio, il proprietario del gruppo di accesso è diverso da quello della politica. Se il proprietario della politica corrisponde a UserGroupOwner, è possibile omettere questo elemento.
- ActionGroupName è il gruppo di azioni al quale si applica la politica.
- ResourceGroupName è il gruppo di risorse al quale si applica la politica.

Aggiunta del controllo accessi al negozio

Dalla prospettiva dello sviluppo, i tipi più comuni di controllo accessi richiesti sono per le nuove viste e comandi creati per il negozio. Tuttavia, è possibile aggiungere altri tipi di controllo accessi al negozio. Per ulteriori informazioni sul controllo accessi per le viste, i comandi e altre funzioni, consultare il manuale the *IBM WebSphere Commerce Access Control Guide*. Prima di continuare con le successive operazioni descritte in questa guida, rivedere il manuale *IBM WebSphere Commerce Access Control Guide*.

Se si aggiungono funzioni di controllo accessi a un negozio basato su un esempio, modificare il file XML delle politiche di controllo accessi di alto livello esistente. Per istruzioni dettagliate, fare riferimento a “Modifica dei file di controllo accessi nell’archivio di esempio” a pagina 201. Se si aggiunge il controllo accessi a un negozio non basato su un esempio, sarà necessario creare un nuovo file XML delle politiche di controllo accessi di alto livello e poi convertirlo. Per istruzioni dettagliate, fare riferimento a “Creazione del controllo accessi nel negozio”.

Creazione del controllo accessi nel negozio

Le risorse del controllo accessi sono diverse dalle altre risorse nel negozio perché per il controllo accessi l’utente crea due file XML delle politiche di controllo accessi di alto livello e poi li converte. I file XML risultanti possono essere caricati nel database mediante il pacchetto Loader o utilizzati nell’archivio di esempio.

Per creare le risorse di controllo accessi, procedere come segue:

1. Rivedere i file XML di alto livello utilizzati per creare le risorse del negozio relativi ai negozi di esempio: *nomenegozioesempioAccessPolicies.xml* e *nomenegozioesempioAccessPolicies_locale.xml*. Questi file si trovano nella seguente directory:
 -  unità:\WebSphere\CommerceServer\samples\stores*nomenegozioesempio*
 -  unità:\ProgramFiles\WebSphere\CommerceServer\samples\stores*nomenegozioesempio*
 -  /usr/WebSphere/CommerceServer/samples/stores/*nomenegozioesempio*
 -  /opt/WebSphere/CommerceServer/samples/stores/*nomenegozioesempio*
 -  /opt/WebSphere/CommerceServer/samples/stores/*nomenegozioesempio*
 -  /QIBM/ProdData/WebCommerce/samples/stores/*nomenegozioesempio* dove *nomenegozioesempio* è il nome dell’archivio negozio di esempio su cui si basa il negozio, ad esempio NewFashion.
2. Fare riferimento alle informazioni nell’Appendice B, “Creazione di dati” a pagina 319.
3. Creare un file *nomenegozioAccessPolicies.xml*, copiando uno dei file *nomenegozioesempioAccessPolicies.xml* oppure creandone uno nuovo. Per ulteriori informazioni, vedere il file DTD che corrisponde a *nomenegozioesempioAccessPolicies.xml*. I file DTD si trovano nelle seguenti directory:

-  unità:\WebSphere\CommerceServer\xml\policies\dtd
 -  unità:\Program Files\WebSphere\CommerceServer\xml\policies\dtd
 -  /usr/WebSphere/CommerceServer/xml/policies/dtd
 -  /opt/WebSphere/CommerceServer/xml/policies/dtd
 -  /opt/WebSphere/CommerceServer/xml/policies/dtd
 -  /QIBM/ProdData/WebCommerce/xml/policies/dtd
4. Creare un file XML di alto livello diverso per ciascuna locale supportata dal negozio. Il file relativo alla locale deve specificare tutte le descrizioni e contenere le informazioni sulla descrizione e sul nome, in modo che possa essere facilmente tradotto. Creare questo file, *nomenegozioAccessPolicies_locale.xml* per ciascuna lingua del negozio copiando uno dei file *nomenegozioesempioAccessPolicies_locale.xml* o creandone uno nuovo.

Nota: Sarà necessario creare un file specifico della locale anche se il negozio supporta solo una lingua.

5. Aggiungere al file le informazioni sul controllo accessi. Per ulteriori informazioni, consultare “Controllo accessi nei negozi di esempio” a pagina 194 e il manuale *IBM WebSphere Commerce Access Control Guide*.
6. Copiare *nomenegozioAccessPolicies.xml* e *nomenegozioAccessPolicies_locale.xml* nella seguente directory:
-  unità:\WebSphere\CommerceServer\xml\policies\xml
 -  unità:\Program Files\WebSphere\CommerceServer\xml\policies\xml
 -  /usr/WebSphere/CommerceServer/xml/policies/xml
 -  /opt/WebSphere/CommerceServer/xml/policies/xml
 -  /opt/WebSphere/CommerceServer/xml/policies/xml
 -  Eeguire la copia in una qualsiasi directory di dati utente. Specificare il percorso completo della DTD nei file XML esistenti. I file DTD di controllo accessi si trovano nella seguente directory:
/QIBM/ProdData/WebCommerce/xml/policies/dtd
7. Eseguire il comando `xmltransform` per convertire *nomenegozioAccessPolicies.xml*.
- a. Dal prompt dei comandi, passare alla seguente directory:
-  unità:\WebSphere\CommerceServer\bin
 -  unità:\ProgramFiles\WebSphere\CommerceServer\bin
 -  /usr/WebSphere/CommerceServer\bin
 -  /opt/WebSphere/CommerceServer/bin
 -  /opt/WebSphere/CommerceServer/bin
- b. Quindi immettere: `xmltransform -infile ..\xml\policies\xml\samplestorenameAccessPolicies.xml -transform ..\xml\policies\xsl\accesscontrol.xsl -outfile ..\xml\policies\xml\samplestorenameAccessPoliciesOut.xml`

```

▶ 400 TRNWCSXML INFILE (input file)
TRANSFORM('/QIBM/ProdData/WebCommerce/xml/policies/xsl
/accesscontrol.xsl') INSTROOT(instance_root) OUTFILE(output_file)

```

- c. Controllare il seguente file di log per verificare che la conversione sia avvenuta correttamente:
- ▶ NT unità:\WebSphere\CommerceServer\bin\xmltransform.db2.log
 - ▶ 2000 unità:\Program Files\WebSphere\CommerceServer\bin\xmltransform.db2.log
 - ▶ AIX /usr/WebSphere/CommerceServer/bin/xmltransform.db2.log
 - ▶ Solaris /opt/WebSphere/CommerceServer/bin/xmltransform.db2.log
 - ▶ Linux /opt/WebSphere/CommerceServer/bin/xmltransform.db2.log
 - ▶ 400 /QIBM/UserData/WebCommerce/instances/instancename /logs/TRNWCSXML.tx

Se la conversione è stata eseguita correttamente, viene visualizzato il seguente messaggio: "<DATA> <ORA> java.lang.Class main XMLTransformer Transform Successful"

8. Eseguire il comando `xmltransform` per convertire `nomenegozioAccessPolicies_locale.xml`.
- a. Dal prompt dei comandi, passare alla seguente directory:
- ▶ NT unità:\WebSphere\CommerceServer\bin
 - ▶ 2000 unità:\ProgramFiles\WebSphere\CommerceServer\bin
 - ▶ AIX /usr/WebSphere/CommerceServer\bin
 - ▶ Solaris /opt/WebSphere/CommerceServer/bin
 - ▶ Linux /opt/WebSphere/CommerceServer/bin
- b. A questo punto immettere: `xmltransform -infile ..\xml\policies\xml\nomenegozioAccessPolicies_locale.xml -transform ..\xml\policies\xsl\accesscontrolnls.xsl -outfile ..\xml\policies\xml\nomenegozioAccessPoliciesOut_locale.xml`
- c. ▶ 400 TRNWCSXML INFILE(input file)
TRANSFORM('/QIBM/ProdData/WebCommerce/xml/policies/xsl/accesscontrolnls.xsl') INSTROOT(instance_root)
OUTFILE(output_file)

9. Apportare le seguenti modifiche ai file XML risultanti:
- a. In `nomenegozioAccessPoliciesOut.xml`, sostituire le tag di apertura e chiusura con:
- ```

<?xml version="1.0"?>
<!DOCTYPE accesscontrol-asset SYSTEM "accesscontrol.dtd">
<accesscontrol-asset>
</accesscontrol-asset>

```
- b. In `nomenegozioAccessPoliciesOut_locale.xml`, sostituire le tag di chiusura e apertura con:
- ```

<?xml version="1.0" encoding="correct language code for the file"?>
<!DOCTYPE accesscontrol-asset SYSTEM "../accesscontrol.dtd">
<accesscontrol-asset>
</accesscontrol-asset>

```

- c. In *nomenegozioAccessPoliciesOut_locale.xml*, sostituire @locale con &locale; ad esempio sostituire LANGUAGE_ID="@en_US" con LANGUAGE_ID="&en_US;"
 - d. In *nomenegozioAccessPoliciesOut_locale.xml*, individuare il riferimento alla tabella "acpoldesc". Rimuovere il simbolo @ alla fine del valore ACPOLICY_ID. Ad esempio, sostituire "@AllUsersExecuteInFashionAllUsersViews@" con "@AllUsersExecuteInFashionAllUsersViews".
 - e. In *nomenegozioAccessPoliciesOut.xml*, sostituire MEMBER_ID="MEMBER_ID" con MEMBER_ID="&MEMBER_ID;"
 - f. In *nomenegozioAccessPoliciesOut.xml* individuare il riferimento alla tabella "acpolicy". Rimuovere "@MEMBER_ID" alla fine del valore di ACPOLICY_ID. Ad esempio, sostituire "@AllUsersExecuteInFashionAllUsersViews@MEMBER_ID" con "@AllUsersExecuteInFashionAllUsersViews"
10. A questo punto, esistono due possibilità:
- Caricare i dati di controllo accessi mediante il pacchetto Loader. Per ulteriori informazioni, consultare la sezione Parte 7, "Pubblicazione del negozio" a pagina 213.
 - Aggiungere i dati di controllo accessi all'archivio negozio. Per ulteriori informazioni sulla creazione di un archivio negozio, consultare Parte 6, "Organizzazione del negozio" a pagina 205.



Per ulteriori informazioni sull'uso dei simboli @ e &, fare riferimento all'Appendice B, "Creazione di dati" a pagina 319.

Modifica dei file di controllo accessi nell'archivio di esempio

WebSphere Commerce fornisce due file XML di controllo accessi pre-convertiti, uno che si applica a tutte le lingue (*nomenegozioesempioAccessPolicies.xml*) e un altro che contiene le informazioni specifiche della locale (*nomenegozioesempioAccessPolicies_locale.xml*), per ciascun negozio. E' necessario convertire entrambi i file in modo da generare altri due file XML, uno che si applica a tutte le lingue (*nomenegozioesempioAccessPoliciesOut.xml*) e l'altro contenente le informazioni specifiche della locale (*nomenegozioesempioAccessPoliciesOut_locale.xml*)

Per modificare la risorsa database del controllo accessi nell'archivio negozio, procedere come segue:

1. Individuare i file XML di controllo precedentemente convertiti per il negozio di esempio su cui si basa il negozio. Questi file vengono denominati *nomenegozioAccessPolicies.xml* e *nomenegozioAccessPolicies_locale.xml*. I file sono ubicati per impostazione predefinita nella seguente directory:
 - **NT** unità:\WebSphere\CommerceServer\samples\stores\nomenegozio
 - **2000** unità:\ProgramFiles\WebSphere\CommerceServer\samples\stores\nomenegozioesempio
 - **AIX** /usr/WebSphere/CommerceServer/samples/stores/nomenegozioesempio
 - **Solaris** /opt/WebSphere/CommerceServer/samples/stores/nomenegozioesempio

- **Linux** /opt/WebSphere/CommerceServer/samples/stores/*nomenegozioesempio*
- **400** /QIBM/ProdData/WebCommerce/samples/stores/*nomenegozioesempio* dove *nomenegozioesempio* è il nome dell'archivio negozio di esempio su cui si basa il negozio, ad esempio NewFashion.

Nota: La modifica dei file DTD corrispondenti può rendere inutilizzabili le politiche.

- Apportare le modifiche necessarie al file. Per ulteriori informazioni sul file esistente, consultare "Comprensione dei file delle politiche di controllo accessi dei negozi di esempio" a pagina 195. Per ulteriori informazioni sui metodi di controllo accessi disponibili in WebSphere Commerce, consultare il manuale *IBM WebSphere Commerce Access Control Guide*.
- Copiare *nomenegozioesempioAccessPolicies.xml* e *nomenegozioesempioAccessPolicies_locale.xml* nella seguente directory:
 - **NT** unità:\WebSphere\CommerceServer\xml\policies\xml
 - **2000** unità:\Program Files\WebSphere\CommerceServer\xml\policies\xml
 - **AIX** /usr/WebSphere/CommerceServer/xml/policies/xml
 - **Solaris** /opt/WebSphere/CommerceServer/xml/policies/xml
 - **Linux** /opt/WebSphere/CommerceServer/xml/policies/xml
 - **400** Eseguire la copia in una qualsiasi directory di dati utente. Specificare il percorso completo della DTD nei file XML esistenti. I file DTD di controllo accessi si trovano nella seguente directory: /QIBM/ProdData/WebCommerce/xml/policies/dtd
- Eseguire il comando `xmltransform` per convertire *nomenegozioesempioAccessPolicies.xml*.
 - Dal prompt dei comandi, passare alla seguente directory:
 - **NT** unità:\WebSphere\CommerceServer\bin
 - **2000** unità:\ProgramFiles\WebSphere\CommerceServer\bin
 - **AIX** /usr/WebSphere/CommerceServer\bin
 - **Solaris** /opt/WebSphere/CommerceServer/bin
 - **Linux** /opt/WebSphere/CommerceServer/bin
 - Quindi immettere:
`xmltransform -infile ..\xml\policies\xml\samplestorenameAccessPolicies -transform ..\xml\policies\xsl\accesscontrol.xsl -outfile ..\xml\policies\xml\samplestorenameAccessPoliciesOut.xml`
400 `TRANSFORM('/QIBM/ProdData/WebCommerce/xml/policies/xsl/accesscontrol.xsl')`
`INSTROOT(instance_root) OUTFILE(output_file)`
 - Controllare il seguente file di log per verificare che la conversione sia avvenuta correttamente:
 - **NT** unità:\WebSphere\CommerceServer\bin\xmltransform.db2.log
 - **2000** unità:\Program Files\WebSphere\CommerceServer\bin\xmltransform.db2.log

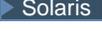
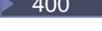
-  /usr/WebSphere/CommerceServer/bin/xmltransform.db2.log
-  /opt/WebSphere/CommerceServer/bin/xmltransform.db2.log
-  /opt/WebSphere/CommerceServer/bin/xmltransform.db2.log
-  /QIBM/UserData/WebCommerce/instances/
instancename/logs/TRNWCXML.tx

Se la conversione è stata eseguita correttamente, viene visualizzato il seguente messaggio: "<DATA> <ORA> java.lang.Class main XMLTransformer Transform Successful"

- Eeguire il comando `xmltransform` per convertire `nomenegozioesempioAccessPolicies_locale.xml`.
 - Dal prompt dei comandi, passare alla seguente directory:
 -  unità:\WebSphere\CommerceServer\bin
 -  unità:\ProgramFiles\WebSphere\CommerceServer\bin
 -  /usr/WebSphere/CommerceServer\bin
 -  /opt/WebSphere/CommerceServer/bin
 -  /opt/WebSphere/CommerceServer/bin
 - Quindi immettere: `xmltransform -infile`
`..\xml\policies\xml\samplestorenameAccessPolicies_locale.xml`
`-transform ..\xml\policies\xsl\accesscontrolnls.xsl -outfile`
`..\xml\policies\xml\samplestorenameAccessPoliciesOut_locale.xml`
 -  `TRNWCXML INFILE(input file)`
`TRANSFORM(' /QIBM/ProdData/WebCommerce/xml/policies/`
`xsl/accesscontrolnls.xsl')`
`INSTROOT(instance_root) OUTFILE(output_file)`
- Apportare le seguenti modifiche ai file XML risultanti:
 - In `nomenegozioesempioAccessPoliciesOut.xml`, sostituire le tag di apertura e chiusura con:


```
<?xml version="1.0"?>
<!DOCTYPE accesscontrol-asset SYSTEM "accesscontrol.dtd">
<accesscontrol-asset>
</accesscontrol-asset>
```
 - In `nomenegozioesempioAccessPoliciesOut_locale.xml`, sostituire le tag di apertura e chiusura con:


```
<?xml version="1.0" encoding="correct language code for the file"?>
<!DOCTYPE accesscontrol-asset SYSTEM "../accesscontrol.dtd">
<accesscontrol-asset>
</accesscontrol-asset>
```
 - In `nomenegozioesempioAccessPoliciesOut_locale.xml`, sostituire `@locale` con `&locale`; ad esempio sostituire `LANGUAGE_ID="@en_US"` con `LANGUAGE_ID="&en_US;"`
 - In `nomenegozioesempioAccessPoliciesOut_locale.xml`, individuare il riferimento alla tabella "acpoldesc". Rimuovere il simbolo `@` alla fine del valore `ACPOLICY_ID`. Ad esempio, sostituire `"@AllUsersExecuteInFashionAllUsersViews@"` con `"@AllUsersExecuteInFashionAllUsersViews"`.
 - In `nomenegozioesempioAccessPoliciesOut.xml`, sostituire `MEMBER_ID="MEMBER_ID"` con `MEMBER_ID="&MEMBER_ID;"`

- f. In *nomenegozioesempioAccessPoliciesOut.xml* individuare il riferimento alla tabella "acpolicy". Rimuovere "@MEMBER_ID" alla fine del valore di ACPOLICY_ID. Ad esempio, sostituire "@AllUsersExecuteInFashionAllUsersViews@MEMBER_ID" con "@AllUsersExecuteInFashionAllUsersViews"
7. Individuare il file dell'archivio negozio per il negozio, (ad esempio, mystore.sar). I file dell'archivio negozio si trovano, per impostazione predefinita, nella seguente directory:
-  NT unità:\WebSphere\CommerceServer\instances\instancename\sar
 -  2000 unità:\Program Files\WebSphere\CommerceServer\instances\instancename\sar
 -  AIX /usr/WebSphere/CommerceServer/instances/instancename/sar
 -  Solaris /opt/WebSphere/CommerceServer/instances/instancename/sar
 -  Linux /opt/WebSphere/CommerceServer/instances/instancename/sar
 -  400 /QIBM/UserData/WebCommerce/instances/instancename/sar
8. Rinominare *nomenegozioesempioAccessPoliciesOut.xml* e *nomenegozioesempioAccessPoliciesOut_locale.xml* in *accesscontrol.xml*
- Nota:** Il file *accesscontrol.xml* specifico per la locale è ubicato per impostazione predefinita nella directory dati/locale, ad esempio data/en_US.
9. Aprire il file dell'archivio negozio utilizzando un programma ZIP.
10. Sostituire i file *accesscontrol.xml* e *accesscontrol.xml* specifico della locale con quelli rinominati al passo 8.
11. Salvare il file dell'archivio negozio.

Parte 6. Organizzazione del negozio

Capitolo 25. Organizzazione di un negozio

Se si desidera utilizzare il negozio come esempio, sviluppare il negozio su un altro server o piattaforma oppure utilizzarlo come base per la creazione di altri negozi, è possibile organizzare il negozio nel formato di archivio negozio.

I negozi di esempio forniti con WebSphere Commerce organizzati sono archivi negozio. I file degli archivi negozio di esempio sono raggruppati come segue:

- Risorse Web: I file utilizzati per creare le pagine del negozio, quali file HTML, file JSP, immagini, grafici e file di inclusione. Le risorse Web sono raggruppate in un file compresso (`webapp.zip`) nell'archivio negozio.
- Bundle di risorse proprietà (facoltative): Contengono il testo per le pagine del negozio. Se il negozio supporta più di una lingua, il bundle di risorse conterrà più bundle, uno per lingua. I bundle delle risorse property sono raggruppati in un file compresso (`properties.zip`) nell'archivio negozio.
- Risorse dati del negozio: I dati da caricare nel database. Le risorse dati del negozio includono dati quali informazioni sulle campagne, i cataloghi, le valute, l'evasione ordini e informazioni sull'assegnazione dei prezzi, la spedizione, il negozio e la tassazione.
- Risorse pagamento: Informazioni sulla configurazione per IBM Payment Manager.
- Descrittore: Un file XML, `sarinfo.xml`, che descrive l'archivio negozio, inclusi i nomi dei file di archivio compressi delle risorse Web, i bundle di risorse e i file XML delle risorse database del negozio. Il file `sarinfo.xml` contiene anche i nomi dei file di inclusione e dei file di controllo congruenza, nonché le informazioni sul file di archivio necessarie durante il processo di pubblicazione. Il file `sarinfo.xml` è l'unico file obbligatorio in un archivio negozio.

Creazione di un archivio del negozio

Per organizzare il negozio come un archivio negozio, attenersi alla seguente procedura:

1. Rivedere la struttura e il contenuto degli archivi negozio di esempio forniti con WebSphere Commerce.

I file degli archivi negozio si trovano nelle seguenti directory:

-  `NT` unità:\WebSphere\CommerceServer\samplstores
-  `2000` unità:\Program Files\WebSphere\CommerceServer\samplstores
-  `AIX` `/usr/WeSphere/CommerceServer/samplstores`
-  `Solaris` `/opt/WebSphere/CommerceServer/samplstores`
-  `Linux` `/opt/WebSphere/CommerceServer/samplstores`
-  `400` `/qibm/proddata/WebCommerce/samplstores`

Per visualizzare l'archivio negozio, utilizzare un programma di decompressione.

2. Creare una directory temporanea in WebSphere Commerce Server per il negozio. Ad esempio, *mystore*.

3. Raggruppare le risorse Web (file JSP, HTML e immagini) in una directory chiamata *webapp*, nella directory temporanea. Creare un file di archivio compresso, utilizzando un programma ZIP, della cartella *webapp*.
4. (Facoltativo) Raggruppare i bundle di risorse *property* in una directory chiamata *properties* della directory temporanea. Se il negozio supporta più di una lingua, il bundle di risorse conterrà più bundle, uno per lingua. Creare un file di archivio compresso, utilizzando un programma ZIP, della cartella *properties*.
5. Raggruppare le risorse dati del negozio in una directory chiamata *data* della directory temporanea. Se il negozio supporta varie lingue, creare sottodirectory per le informazioni specifiche sulle lingue utilizzando i nomi di locale. Ad esempio, *en_US*.
6. (Facoltativo) Copiare il file *sarrule.xml* da un archivio negozio esistente nella directory *data*. Il file *sarrule.xml* si trova nella directory *data* degli archivi negozio di esempio. Il file *sarrule.xml* agisce come controllore della coerenza quando si utilizza Servizi del negozio per eseguire una pubblicazione. Per ulteriori informazioni sul file *sarrule*, fare riferimento all'Appendice D, "sarrule.xml" a pagina 327.
7. Creare una directory chiamata *SAR-INF* nella directory temporanea.
8. Creare un file *sarinfo.xml* per l'archivio negozio. Per ulteriori informazioni sulle specifiche XML, fare riferimento al file di descrizione dell'archivio, *sarinfo.dtd* nella seguente directory:

-  unità:\WebSphere\CommerceServer\xml\sar
-  unità:\Program Files\WebSphere\CommerceServer\xml\sar
-  /usr/WebSphere/CommerceServer/xml/sar
-  /opt/WebSphere/CommercServer/xml/sar
-  /opt/WebSphere/CommercServer/xml/sar
-  /qibm/proddata/WebCommerce/xml/sar

- a. Utilizzando l'esempio e le informazioni contenute nell'Appendice C, "sarinfo.xml" a pagina 321 come guida, creare un file *sarinfo.xml*. L'ordine in cui vengono pubblicate le risorse dati è importante, dal momento che le risorse dati devono essere pubblicate prima delle altre risorse. Quindi, l'ordine delle risorse specificato nel file *sarinfo.xml*, deve corrispondere all'ordine delle risorse specificato nei file *sarinfo.xml* per i negozi di esempio.

Nota: Se si sceglie di includere un file *sarrule.xml* nell'archivio negozio (vedere passo 6), è necessario includere le informazioni su *sarrule.xml* nel file *sarinfo.xml*. Se si sceglie di non includerlo, assicurarsi non venga citato che nel file *sarinfo.xml*.

- b. Salvare il file *sarinfo.xml* nella directory *SAR-INF* creata al passo 6.
9. Creare un file ZIP composto dal file ZIP delle risorse Web, dai bundle delle risorse *property*, dalle risorse dati del negozio e dalla directory *sar-inf*. Denominare questo file ZIP *nomearchivionegozio.sar*.
10. Se si desidera modificare o pubblicare l'archivio negozio utilizzando Servizi del negozio, salvare il file *nomearchivionegozio.sar* nella seguente directory:

-  unità:\WebSphere\CommerceServer\instances*nome istanza*\sar
-  unità:\Program Files\WebSphere\CommerceServer\instances*nome istanza*\sar

-  /usr/WebSphere/CommerceServer/instances/*nomeistanza*/sar
 -  /opt/WebSphere/CommerceServer/instances/*nomeistanza*/sar
 -  /opt/WebSphere/CommerceServer/instances/*nomeistanza*/sar
 -  /QIBM/UserData/WebCommerce/instances/*nomeistanza*/sar
11. L'archivio negozio sarà visualizzato adesso nell'elenco degli archivi negozio di Servizi del negozio.

Creazione di un archivio negozio di esempio

Una volta organizzato il negozio come archivio negozio, è possibile utilizzarlo come un negozio di esempio di Servizi del negozio. Un archivio negozio di esempio rappresenta un archivio negozio da copiare ed utilizzare come una base su cui creare nuovi negozi. Per utilizzare l'archivio negozio come un archivio negozio di esempio, attenersi alla seguente procedura:

1. Salvare il file di archivio negozio nella seguente directory:
 -  unità:\WebSphere\CommerceServer\samplstores
nomearchivionegozio
 -  unità:\Program Files\WebSphere\CommerceServer\samplstores
nomearchivionegozio
 -  /usr/WeSphere/CommerceServer/samplstores
/*nomearchivionegozio*
 -  /opt/WebSphere/CommerceServer/samplstores
/*nomearchivionegozio*
 -  /opt/WebSphere/CommerceServer/samplstores
/*nomearchivionegozio*
 -  /qibm/proddata/WebCommerce/samplstores
/*nomearchivionegozio*
2. (Facoltativo) Creare pagine di anteprima. Per visualizzare le anteprime delle pagine del negozio in Servizi del negozio, è necessario creare pagine di anteprima. Attenersi alla procedura di seguito riportata:
 - a. (Facoltativo) In Servizi del negozio, selezionare **Nuovo**. Viene visualizzata la pagina Creazione archivio negozio. Dall'elenco **Esempio**, selezionare un negozio di esempio, quindi fare clic su **Anteprima**. Le pagine visualizzate sono chiamate pagine di anteprima. Queste pagine sono file HTML che presentano un flusso di acquisti di esempio predefinito e agiscono come anteprima del negozio di esempio.
 - b. Stabilire il flusso di acquisiti che si desidera visualizzare nelle pagine di anteprima.
 - c. (Facoltativo) Creare alcuni dati di esempio in un negozio pubblicato. Ad esempio, aggiungere degli articoli nel carrello degli acquisti e creare alcuni indirizzi di spedizione e di fatturazione. Verranno create delle pagine di anteprima di questo negozio e i dati renderanno le pagine più realistiche.
 - d. Utilizzando Internet Explorer, esaminare il negozio. Salvare il file HTML per ciascuna pagina, selezionando File, Salva con nome. Salvare anche il foglio di stile (.css) e le immagini. Salvare i file nelle seguenti directory:
 - *stylesheet.css*

- **NT** unità:\WebSphere\AppServer\installedApps\WC_Enterprise_App_nomeistanza.ear\wcstools.war\tools\devtools\nomearchivionegozio\preview
- **2000** unità:\Program Files\WebSphere\AppServer\installedApps\WC_Enterprise_App_nomeistanza.ear\wcstools.war\tools\devtools\nomearchivionegozio\preview
- **AIX** /usr/WebSphere/AppServer/installedApps/WC_Enterprise_App_nomeistanza.ear/wcstools.war/tools/devtools/nomearchivionegozio/preview
- **Solaris** /opt/WebSphere/AppServer/installedApps/WC_Enterprise_App_nomeistanza.ear/wcstools.war/tools/devtools/nomearchivionegozio/preview
- **Linux** /opt/WebSphere/AppServer/installedApps/WC_Enterprise_App_nomeistanza.ear/wcstools.war/tools/devtools/nomearchivionegozio/preview
- **400** /Qibm/UserData/WebAsAdv4/WASnomeistanza/installedApps/WC_Enterprise_App_nomeistanza.ear/wcstools.war/tools/devtools/nomearchivionegozio/preview
- HTML
 - **NT** unità:\WebSphere\AppServer\installedApps\WC_Enterprise_App_nomeistanza.ear\wcstools.war\tools\devtools\nomearchivionegozio\preview\locale
 - **2000** unità:\Program Files\WebSphere\AppServer\installedApps\WC_Enterprise_App_nomeistanza.ear\wcstools.war\tools\devtools\nomearchivionegozio\preview\locale
 - **AIX** /usr/WebSphere/AppServer/installedApps/WC_Enterprise_App_nomeistanza.ear/wcstools.war/tools/devtools/nomearchivionegozio/preview/locale
 - **Solaris** /opt/WebSphere/AppServer/installedApps/WC_Enterprise_App_nomeistanza.ear/wcstools.war/tools/devtools/nomearchivionegozio/preview/locale
 - **Linux** /opt/WebSphere/AppServer/installedApps/WC_Enterprise_App_nomeistanza.ear/wcstools.war/tools/devtools/nomearchivionegozio/preview/locale
 - **400** /Qibm/UserData/WebAsAdv4/WASnomeistanza/installedApps/WC_Enterprise_App_nomeistanza.ear/wcstools.war/tools/devtools/nomearchivionegozio/preview/locale
- immagini indipendenti dalla locale
 - **NT** unità:\WebSphere\AppServer\installedApps\WC_Enterprise_App_nomeistanza.ear\wcstools.war\tools\devtools\nomearchivionegozio\preview\images
 - **2000** unità:\Program Files\WebSphere\AppServer\installedApps\WC_Enterprise_App_nomeistanza.ear\wcstools.war\tools\devtools\nomearchivionegozio\preview\images
 - **AIX** /usr/WebSphere/AppServer/installedApps/WC_Enterprise_App_nomeistanza.ear/wcstools.war/tools/devtools/nomearchivionegozio/preview/images

- **Solaris** /opt/WebSphere/AppServer/installedApps/WC_Enterprise_App_nomeistanza.ear/wcstools.war/tools/devtools/nomearchivionegozio/preview/images
 - **Linux** /opt/WebSphere/AppServer/installedApps/WC_Enterprise_App_nomeistanza.ear/wcstools.war/tools/devtools/nomearchivionegozio/preview/images
 - **400** /Qibm/UserData/WebAsAdv4/WASnomeistanza/installedApps/WC_Enterprise_App_nomeistanza.ear/wcstools.war/tools/devtools/nomearchivionegozio/preview/images
- immagini dipendenti dalla locale
 - **NT** unità:\WebSphere\AppServer\installedApps\WC_Enterprise_App_nomeistanza.ear\wcstools.war\tools\devtools\nomearchivionegozio\preview\locale\images
 - **2000** unità:\Program Files\WebSphere\AppServer\installedApps\WC_Enterprise_App_nomeistanza.ear\wcstools.war\tools\devtools\nomearchivionegozio\preview\locale\images
 - **AIX** /usr/WebSphere/AppServer/installedApps/WC_Enterprise_App_nomeistanza.ear/wcstools.war/tools/devtools/nomearchivionegozio/preview/locale/images
 - **Solaris** /opt/WebSphere/AppServer/installedApps/WC_Enterprise_App_nomeistanza.ear/wcstools.war/tools/devtools/nomearchivionegozio/preview/locale/images
 - **Linux** /opt/WebSphere/AppServer/installedApps/WC_Enterprise_App_nomeistanza.ear/wcstools.war/tools/devtools/nomearchivionegozio/preview/locale/images
 - **400** /Qibm/UserData/WebAsAdv4/WASnomeistanza/installedApps/WC_Enterprise_App_nomeistanza.ear/wcstools.war/tools/devtools/nomearchivionegozio/preview/locale/images
- e. Poiché l'ubicazione delle immagini e del file css è cambiata, è necessario cambiare anche i riferimenti alle immagini e il file css nelle pagine HTML. Una volta modificati i riferimenti, verificare che le immagini vengano visualizzate quando si aprono le pagine HTML in un browser.
- f. Modificare i link nelle pagine HTML passando dai comandi ai link che fanno riferimento ai file HTML.
3. Creare un file HTML di riepilogo dell'archivio negozio. Queste informazioni vengono visualizzate in **Descrizione esempio** della pagina di creazione dell'archivio negozio di Servizi del negozio.
- a. Utilizzando il seguente esempio come guida, creare il nuovo file.
- ```
<doctype html public "-//w3c//dtd html 4.0 transitional//en">
<html>
<head>
 <meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
</head>
<body>
 Descrivere il negozio
</body>
</html>
```

b. Salvare questo file come `Feature_locale.html`, dove `locale` è l'abbreviazione per la lingua utilizzata. Ad esempio, `en_US`. Salvare questo file nella seguente directory:

- ▶ **NT** unità: `\WebSphere\CommerceServer\samplestores\nomearchivionezio`
- ▶ **2000** unità: `\Program Files\WebSphere\CommerceServer\samplestores\nomearchivionezio`
- ▶ **AIX** `/usr/WebSphere/CommerceServer/samplestores/nomearchivionezio`
- ▶ **Solaris** `/opt/WebSphere/CommerceServer/samplestores/nomearchivionezio`
- ▶ **Linux** `/opt/WebSphere/CommerceServer/samplestores/nomearchivionezio`
- ▶ **400** `/qibm/proddata/WebCommerce/samplestores/nomearchivionezio`

4. Aggiungere l'archivio negozio al file `sarregistry.xml`. Il file `sarregistry.xml` determina gli archivi negozio visualizzati nell'elenco **Esempio** della pagina di creazione dell'archivio negozio di Servizi del negozio. Il file `sarregistry.xml` determina inoltre le pagine di anteprima e il file delle funzioni associati a ciascun archivio negozio. `sarregistry.xml` si trova nella seguente directory:

- ▶ **NT** unità: `\WebSphere\CommerceServer\xml\tools\devtools`
- ▶ **2000** unità: `\Program Files\WebSphere\CommerceServer\xml\tools\devtools`
- ▶ **AIX** `/usr/WebSphere/CommerceServer/xml/tools/devtools`
- ▶ **Solaris** `/opt/WebSphere/CommerceServer/xml/tools/devtools`
- ▶ **Linux** `/opt/WebSphere/CommerceServer/xml/tools/devtools`
- ▶ **400** `/QIBM/ProdData/WebCommerce/xml/tools/devtools`

a. Utilizzando il seguente esempio come guida, aggiungere il nuovo archivio negozio al file `sarregistry.xml`.

```
<SampleSAR fileName="infashion_en_US_es_ES.sar" relativePath="InFashion">
 <html locale="es_ES" featureFile="InFashion/Feature_es_ES.html"
 sampleSite="RetailModel/preview/es_ES/index.html"/>
 <html locale="en_US" featureFile="InFashion/Feature_en_US.html"
 sampleSite="RetailModel/preview/en_US/index.html"/>
</SampleSAR>
```

Questo esempio definisce le pagine di anteprima per gli archivi negozio `InFashion` inglese e spagnolo. Le righe in grassetto definiscono la home page per le pagine di anteprima inglesi e spagnole.

5. L'archivio negozio dovrebbe ora essere visualizzato nell'elenco **Esempio** della pagina di creazione dell'archivio negozio di Servizi del negozio.

---

## Parte 7. Pubblicazione del negozio

Per creare un negozio funzionante, le risorse Web della facciata del negozio devono essere pubblicate su WebSphere Commerce Server e i dati del negozio sul database di WebSphere Commerce.

I capitoli di questa sezione illustrano le opzioni relative alla pubblicazione fornite da WebSphere Commerce:

- Capitolo 26, "Pubblicazione di un negozio completo" a pagina 215 - Questo capitolo illustra la pubblicazione di un intero negozio (facciata e risorse dati) nel formato di un archivio di negozio con Servizi del negozio o la riga comandi.
- Capitolo 27, "Panoramica sul caricamento dei dati" a pagina 229 - Questo capitolo illustra la pubblicazione delle risorse dati del negozio nel database mediante il pacchetto Loader e altri componenti di WebSphere Catalog Manager.
- Capitolo 28, "Caricamento dei gruppi di risorse di database di WebSphere Commerce" a pagina 271 - Questo capitolo illustra la pubblicazione di gruppi di risorse dati del negozio o di tutti i dati del negozio nel database mediante il pacchetto Loader e altri componenti di WebSphere Catalog Manager.
- Capitolo 29, "Pubblicazione dei contratti e dei conti commerciali" a pagina 285 - Questo capitolo illustra la pubblicazione delle risorse gruppi di prodotti, contratti e conti.
- Capitolo 30, "Pubblicazione delle risorse della facciata e dei file di configurazione del negozio" a pagina 289 - Questo capitolo illustra la pubblicazione dei file di configurazione e delle risorse della facciata del negozio.



---

## Capitolo 26. Pubblicazione di un negozio completo

Per creare un negozio funzionante, le risorse Web della facciata del negozio devono essere pubblicate sul WebSphere Commerce Server e i dati del negozio sul database di WebSphere Commerce. Questo capitolo illustra la pubblicazione di un intero negozio (facciata e risorse dati) in formato archivio negozio con Servizi del negozio o la riga comandi.

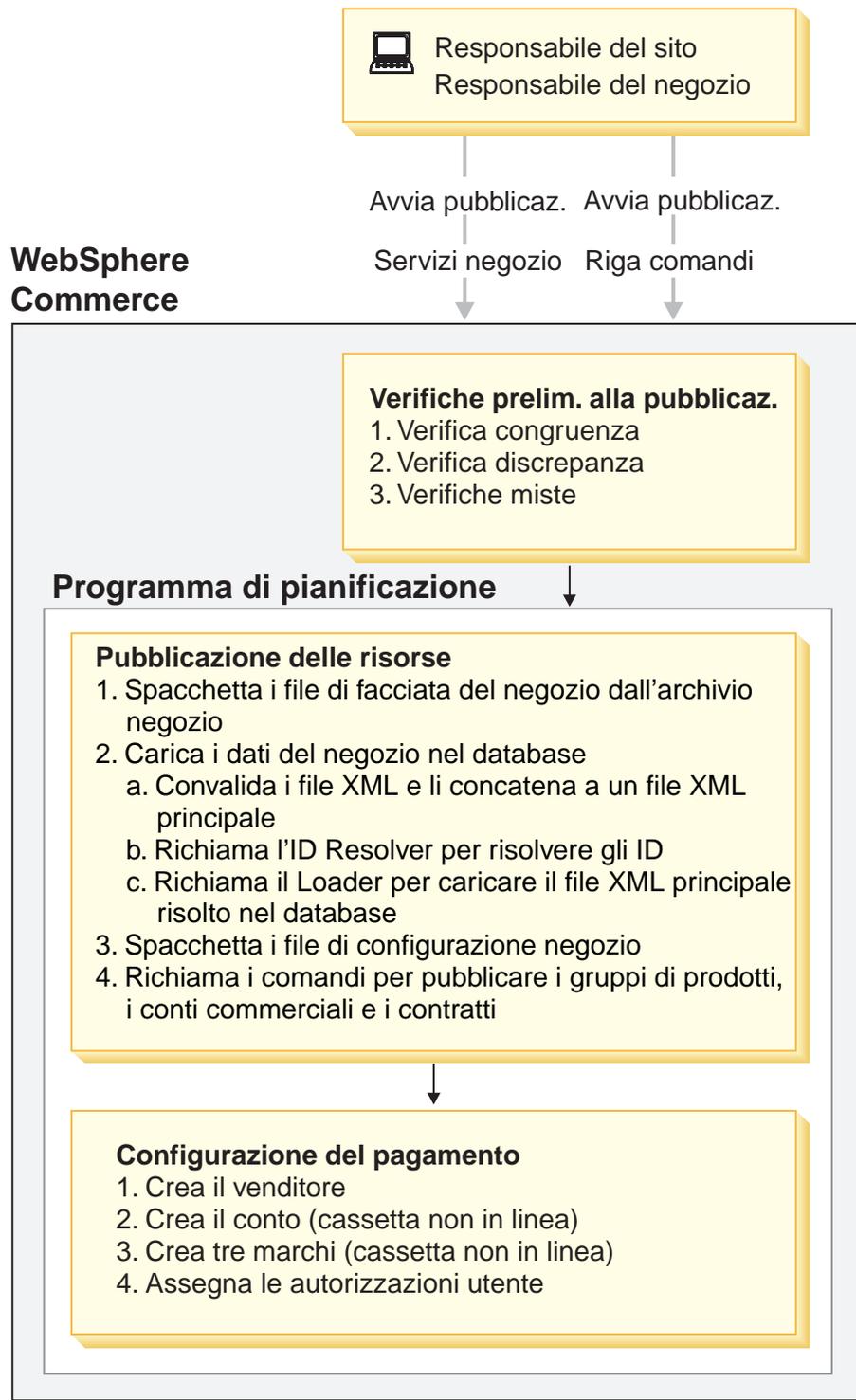
**Nota:** Se si preferisce non creare il negozio come archivio, è possibile pubblicare le risorse singolarmente. Per ulteriori informazioni, consultare Capitolo 27, "Panoramica sul caricamento dei dati" a pagina 229, Capitolo 28, "Caricamento dei gruppi di risorse di database di WebSphere Commerce" a pagina 271, Capitolo 29, "Pubblicazione dei contratti e dei conti commerciali" a pagina 285 e Capitolo 30, "Pubblicazione delle risorse della facciata e dei file di configurazione del negozio" a pagina 289.

---

### Comprensione della pubblicazione in WebSphere Commerce

L'opzione relativa alla pubblicazione disponibile in Servizi del negozio o dalla riga comandi consente di pubblicare un negozio completo (risorse dati e facciata) in una sola volta. Per poter utilizzare questa opzione, le risorse del negozio devono essere impacchettate in modo da formare un archivio. Per ulteriori informazioni sull'impacchettamento di un negozio come archivio, consultare Parte 6, "Organizzazione del negozio" a pagina 205.

Il grafico di seguito riportato illustra la procedura relativa al processo di pubblicazione.



## Avvio della pubblicazione

Per pubblicare un negozio, è necessario avere autorità di responsabile del sito o responsabile del negozio (tutti i negozi). I responsabili del sito o del negozio possono avviare il processo di pubblicazione con uno dei seguenti metodi:

- Servizi del negozio
- Riga comandi

Entrambi i metodi di pubblicazione richiedono la selezione dell'archivio negozio da pubblicare. Con entrambi i metodi, è possibile selezionare quali parti di archivio negozio si desidera pubblicare. Ad esempio, è possibile selezionare una combinazione dei seguenti elementi:

- Risorse database di negozio, con o senza i dati del catalogo in linea
- Risorse Web come file JSP, immagini e file HTML
- File properties (il testo per il negozio)

Alla prima pubblicazione, si consiglia di pubblicare l'intero archivio in modo da poter visualizzare un negozio funzionale. Nelle pubblicazioni successive, comunque, è possibile aggiornare solo uno dei seguenti componenti: risorse database, risorse Web o bundle risorse proprietà.

Se si sceglie di non pubblicare i dati del catalogo in linea la prima volta che si pubblica un archivio negozio, alcune delle risorse dati (contratti e conti) potrebbero non essere pubblicate correttamente perché dipendono dai dati del catalogo. Inoltre, non vengono pubblicati né i dati del catalogo né le risorse Web, l'ID negozio, l'ID catalogo e l'ID lingua non saranno creati e non sarà possibile avviare il negozio da Servizi del negozio. Per ulteriori informazioni, fare riferimento alla sezione "Crea il file parameters.jsp" a pagina 224.

Il catalogo in linea è composto dai seguenti file XML nell'archivio negozio:

- catalog.xml
- offering.xml
- store-catalog.xml
- store-catalog-shipping.xml
- store-catalog-tax.xml
- storefulfill.xml

Se si sceglie di pubblicare queste informazioni in un momento successivo, è possibile pubblicarle con Servizi del negozio o con il pacchetto Loader. Per ulteriori informazioni sul caricamento di un sottoinsieme di dati mediante gruppi di risorse database, ad esempio i dati di gruppo di catalogo sopra elencati, consultare "Caricamento dei gruppi di risorse database" a pagina 279.

Per ulteriori informazioni su come pubblicare un archivio negozio con Servizi del negozio o con la riga comandi, consultare la guida in linea di WebSphere Commerce.

**Nota:** Dopo aver iniziato il processo di pubblicazione con Servizi del negozio o con la riga comandi, non è necessario eseguire alcun'altra operazione. Tutte le altre operazioni descritte nel precedente grafico e in questo capitolo vengono completate dal sistema WebSphere Commerce.

## Verifiche preliminari alla pubblicazione

Una volta che il responsabile del sito o del negozio ha avviato la pubblicazione, WebSphere Commerce, prima di iniziare l'effettivo processo di pubblicazione, esegue diverse verifiche. In particolare, effettua le verifiche di seguito riportate:

- Verifica della coerenza
- Verifica della discrepanza

- Verifiche miste

**Nota:** Il programma di utilità per la pubblicazione della riga comandi verifica i parametri inseriti e completa i controlli di coerenza e discrepanza. Tuttavia, se si esegue la pubblicazione mediante questo programma, non vengono restituiti i messaggi descritti nei seguenti paragrafi. Se la pubblicazione viene invece eseguita mediante la modalità di aggiornamento, la riga comandi sovrascrive il negozio esistente senza visualizzare richieste. Se si utilizza la modalità di inserimento, la riga comandi carica il nuovo negozio. Se la pubblicazione non riesce, viene visualizzato un messaggio di errore.

### **Verifica della coerenza**

La verifica preliminare alla pubblicazione utilizza le regole contenute in questo file per assicurare che le informazioni presenti nei file XML siano coerenti con le risorse Web dell'archivio negozio. Ad esempio, se il file `command.xml` fa riferimento a un particolare file JSP, la verifica assicura che il file JSP si trovi nell'archivio negozio `webapp.zip`. Se la verifica di coerenza rileva un errore, esso viene scritto nel log, ma la pubblicazione continua regolarmente. Per ulteriori informazioni sui file di log, consultare "File di log della pubblicazione" a pagina 226.

Per un esempio di file `sarrule.xml`, consultare Appendice D, "sarrule.xml" a pagina 327.

### **Verifiche di discrepanza**

Durante la verifica preliminare alla pubblicazione, WebSphere Commerce controlla che non vi siano discrepanze tra il negozio e il catalogo. In particolare, verifica nel database i seguenti elementi:

- Un negozio esistente con lo stesso identificativo nella tabella STOREENT: se il negozio esiste già, viene visualizzato un messaggio che richiede se si desidera sovrascrivere il negozio o annullare la pubblicazione.
- Un catalogo esistente con lo stesso identificativo: se il catalogo esiste già, viene visualizzato un messaggio che richiede se si desidera sovrascrivere il catalogo o annullare la pubblicazione. Se il catalogo appartiene ad altri negozi, viene visualizzato un messaggio che indica quali sono i negozi e richiede se si desidera continuare l'operazione e sovrascrivere il catalogo oppure annullare la pubblicazione.

### **Verifiche miste**

Durante la verifica preliminare alla pubblicazione, WebSphere Commerce controlla anche che il programma di pianificazione sia abilitato e che la cache e le tabelle di riepilogo siano disabilitate. In caso contrario, viene visualizzato un messaggio di avviso che indica la funzione erroneamente abilitata o disabilitata.

**Nota:** Il programma di utilità per la pubblicazione della riga comandi non esegue queste verifiche.

**Programma di pianificazione:** Come descritto più dettagliatamente nelle successive sezioni, il programma di pianificazione esegue il lavoro di pubblicazione. Se tale programma è disabilitato, il processo di pubblicazione non può essere eseguito.

**Cache e trigger della cache:** Se durante la pubblicazione la cache resta abilitata, è possibile che i trigger vengano richiamati quando il database viene aggiornato. I trigger della cache possono generare attività di database non necessarie che possono causare un overflow del log delle transazioni del database e avere

ripercussioni sull'esecuzione della pubblicazione. Per disabilitare i trigger della cache, consultare la guida in linea di WebSphere Commerce.

**Tabelle di riepilogo:**  Se le tabelle di riepilogo sono abilitate, esse vengono aggiornate durante la pubblicazione provocando un overflow del log delle transazioni del database e un rallentamento dell'esecuzione delle pubblicazione. Per disabilitare le tabelle di riepilogo, consultare la guida in linea di WebSphere Commerce.

## Publicazione delle risorse

La fase della pubblicazione delle risorse è un lavoro eseguito dal programma di pianificazione. Durante l'esecuzione di questo lavoro, WebSphere Commerce completa le seguenti azioni:

- Spacchetta i file di facciata del negozio dell'archivio negozio
- Carica i dati del negozio dai file XML dell'archivio negozio nel database
- Crea il file `parameters.jsp`
- Spacchetta i file di configurazione del negozio
- Richiama i comandi per pubblicare contratti e conti commerciali
- Aggiorna i componenti del registro

### Spacchettamento dei file di facciata del negozio dall'archivio negozio

Spacchettare le risorse Web dall'archivio negozio in WebSphere Commerce Server è la prima azione da eseguire nella fase di pubblicazione delle risorse. Durante lo spacchettamento dei file dall'archivio negozio, WebSphere Commerce effettua le seguenti operazioni:

#### Spacchetta le risorse Web e le copia nelle seguenti ubicazioni su WebSphere Commerce Server:

- I file JSP, HTML, i file di inclusione, le immagini e i grafici vengono pubblicati nella directory del negozio (*dirnegozio*) nella root del documento di Applicazione Web - negozi:
  -  unità:\WebSphere\AppServer\installedApps\WC\_Enterprise\_App\_*nomeistanza*.ear\wcstores.war\dirnegozio
  -  unità:\Program Files\WebSphere\AppServer\installedApps\WC\_Enterprise\_App\_*nomeistanza*.ear\wcstores.war\dirnegozio
  -  /usr/WebSphere/AppServer/installedApps/WC\_Enterprise\_App\_*nomenegozio*.ear/wcstores.war/dirnegozio
  -  /opt/WebSphere/AppServer/installedApps/WC\_Enterprise\_App\_*nomeistanza*.ear/wcstores.war/dirnegozio
  -  /opt/WebSphere/AppServer/installedApps/WC\_Enterprise\_App\_*nomeistanza*.ear/wcstores.war/dirnegozio
  -  /QIBM/UserData/WebASAdv4/WASinstancename/installedApps/WC\_Enterprise\_App\_*nomeistanza*.ear/wcstores.war/dirnegozio
- I bundle di risorse e i file properties sono pubblicati nel percorso delle proprietà dell'applicazione:
  -  unità:\WebSphere\AppServer\installedApps\WC\_Enterprise\_App\_*nomeistanza*.ear\wcstores.war\WEB-INF\classes\dirnegozio

- ▶ 2000 unità:\Program Files\WebSphere\AppServer\installedApps\WC\_Enterprise\_App\_ *nomeistanza* .ear\wcstores.war\WEB-INF\classes\ *dirnegozio*
- ▶ AIX /usr/WebSphere/AppServer/installedApps/WC\_Enterprise\_App\_ *nomeistanza* .ear/wcstores.war/WEB-INF/classes/ *dirnegozio*
- ▶ Solaris /opt/WebSphere/AppServer/installedApps/WC\_Enterprise\_App\_ *nomeistanza* .ear/wcstores.war/WEB-INF/classes/ *dirnegozio*
- ▶ Linux /opt/WebSphere/AppServer/installedApps/WC\_Enterprise\_App\_ *nomeistanza* .ear/wcstores.war/WEB-INF/classes/ *dirnegozio*
- ▶ 400 /QIBM/UserData/WebASAdv4/WASinstancename/installedApps/WC\_Enterprise\_App\_ *nomeistanza* .ear/wcstores.war/WEB-INF/classes/ *dirnegozio*

### Carica i dati del negozio dai file XML dell'archivio negozio nel database

Mentre vengono caricati i dati dai file XML dell'archivio negozio nel database, WebSphere Commerce effettua le seguenti operazioni:

**Nota:** Solo i file XML del tipo db-load vengono caricati nel database. Il tipo di file è specificato nel file *sarinfo.xml*. Per ulteriori informazioni sul file *sarinfo.xml*, consultare Appendice C, "sarinfo.xml" a pagina 321

### Convalida i file XML dell'archivio negozio e li concatena a un file XML

**principale:** WebSphere Commerce convalida i file XML con i corrispondenti file DTD. I file DTD si trovano nelle seguenti directory:

- ▶ NT unità:\WebSphere\CommerceServer\xml\sar
- ▶ 2000 unità:\Program Files\WebSphere\CommerceServer\xml\sar
- ▶ AIX /usr/WebSphere/CommerceServer/xml/sar
- ▶ Solaris /opt/WebSphere/CommerceServer/xml/sar
- ▶ Linux /opt/WebSphere/CommerceServer/xml/sar
- ▶ 400 /qibm/proddata/WebCommerce/xml/sar

Se i file XML non sono validi, WebSphere Commerce scrive un errore nel log degli errori. Se gli errori presenti nel log superano il numero massimo di errori specificato nella sezione DevTools del file di configurazione di WebSphere Commerce, la pubblicazione del file *nome\_istanza.xml*, (per impostazione predefinita *MaxErrorsInSarXML=1*) non riesce. Il file di configurazione di WebSphere Commerce, *nome\_istanza.xml* risiede nella seguente directory:

- ▶ NT unità:\WebSphere\CommerceServer\instances\*nome\_istanza*\xml\*nome\_istanza.xml*
- ▶ 2000 unità:\Program Files\WebSphere\CommerceServer\instances\*nome\_istanza*\xml\*nome\_istanza.xml*
- ▶ AIX /usr/WebSphere/CommerceServer/instances/*nome\_istanza*/xml/*nome\_istanza.xml*

- **Solaris** /opt/WebSphere/CommerceServer/instances/*nomeistanza*  
/xml/*nome\_istanza.xml*
- **Linux** /opt/WebSphere/CommerceServer/instances/*nomeistanza*  
/xml/*nome\_istanza.xml*
- **400** /QIBM/UserData/WebCommerce/instances/*nomeistanza*  
/xml/*nome\_istanza.xml*

Una volta convalidati i file XML, possono essere concatenati a un file: *nomenegoziomaster.xml*. I file vengono concatenati in base alla priorità specificata nel file *sarinfo.xml*. Per ulteriori informazioni, consultare la sezione Appendice C, “*sarinfo.xml*” a pagina 321. Il file *nomenegoziomaster.xml* risiede nella seguente directory:

- **NT** unità:\WebSphere\CommerceServer\temp\*nomeistanza*\tools\devtools
- **2000** unità:\Program Files\WebSphere\CommerceServer\temp\*nomeistanza*\tools\devtools
- **AIX** /usr/WebSphere/CommerceServer/temp/*nomeistanza*/tools/devtools
- **Solaris** /opt/WebSphere/CommerceServer/temp/*nomeistanza*/tools/devtools
- **Linux** /opt/WebSphere/CommerceServer/temp/*nomeistanza*/tools/devtools
- **400** /QIBM/UserData/WebCommerce/instances/*nomeistanza*  
/temp/tools/devtools

**Richiama ID Resolver per risolvere gli ID:** ID Resolver, un programma di utilità del pacchetto Loader, genera identificativi univoci per gli elementi XML nei file XML dell’archivio negozio. Ad esempio, ID Resolver sostituisce l’alias @ utilizzato nei file XML del negozio di esempio con un valore univoco. Per un esempio della risoluzione alias interno utilizzata nei negozi di esempio, fare riferimento all’Appendice B, “Creazione di dati” a pagina 319.

**Nota:** ID Resolver può anche risolvere gli identificativi per i negozi già pubblicati, quando vengono ripubblicati. Ad esempio, se l’archivio negozio è stato pubblicato una volta e occorre pubblicare di nuovo l’intero archivio o parti di esso, ID Resolver richiama gli identificativi univoci dal database e li utilizza durante il processo di ripubblicazione.

Per ulteriori informazioni su ID Resolver e su altri componenti del pacchetto Loader, consultare Capitolo 27, “Panoramica sul caricamento dei dati” a pagina 229.

Quando Servizi del negozio o la funzione di pubblicazione della riga comandi richiamano ID Resolver, è necessario specificare quale metodo di ID Resolver utilizzare. I metodi di ID Resolver che è possibile utilizzare per elaborare l’immissione sono diversi; è possibile, ad esempio, trattare i dati come se gli identificativi esistessero nei dati originali (metodo update) oppure come se fossero nuovi (metodo load). Il metodo misto viene utilizzato quando alcuni identificativi esistono ed altri no.

**Nota:** Il metodo misto è consigliato per Servizi del negozio. Se si esegue la pubblicazione mediante la riga comandi, è anche possibile specificare il modo misto, soprattutto se non si sa se i dati da pubblicare già esistono nel database.

E’ possibile specificare quale metodo sarà utilizzato con Servizi del negozio o con il programma di utilità per la pubblicazione della riga comandi nel file di configurazione di WebSphere Commerce, *nome\_istanza.xml*. Per impostazione

predefinita, Servizi del negozio utilizza il metodo misto. Per ulteriori informazioni sui metodi di ID Resolver, consultare “Comando ID Resolve” a pagina 234.

Servizi del negozio e la funzione di pubblicazione della riga comandi devono specificare anche un file di personalizzazione da utilizzare con ID Resolver. Se non si specifica un file di personalizzazione nel file di configurazione di WebSphere Commerce, *nome\_istanza.xml*, il codice di pubblicazione utilizza uno di quelli predefiniti: DBConnectionCustomizer o OracleConnectionCustomizer.

### Oracle

Il file di personalizzazione OracleConnectionCustomizer si trova nella seguente directory:

- ▶ **NT** unità:\WebSphere\AppServer\installedApps\WC\_Enterprise\_App\_*nome\_istanza*.ear\properties
- ▶ **2000** unità:\Program Files\WebSphere\AppServer\installedApps\WC\_Enterprise\_App\_*nome\_istanza*.ear\properties
- ▶ **AIX** /usr/WebSphere/AppServer/installedApps/WC\_Enterprise\_App\_*nome\_istanza*.ear/properties
- ▶ **Solaris** /opt/WebSphere/AppServer/installedApps/WC\_Enterprise\_App\_*nome\_istanza*.ear/properties
- ▶ **Linux** /opt/WebSphere/AppServer/installedApps/WC\_Enterprise\_App\_*nome\_istanza*.ear/properties
- ▶ **400** /QIBM/UserData/WebASAdv4/WASinstancename/installedApps/WC\_Enterprise\_App\_*nome\_istanza*.ear/properties

### DB2

Il file DBConnectionCustomizer si trova nel seguente file ZIP:

- ▶ **NT** unità:\WebSphere\AppServer\installedApps\WC\_Enterprise\_App\_*nome\_istanza*.ear\lib\loader\idresgen.zip
- ▶ **2000** unità:\Program Files\WebSphere\AppServer\installedApps\WC\_Enterprise\_App\_*nome\_istanza*.ear\lib\loader\idresgen.zip
- ▶ **AIX** /usr/WebSphere/AppServer/installedApps/WC\_Enterprise\_App\_*nome\_istanza*.ear/lib/loader/idresgen.zip
- ▶ **Solaris** /opt/WebSphere/AppServer/installedApps/WC\_Enterprise\_App\_*nome\_istanza*.ear/lib/loader/idresgen.zip
- ▶ **Linux** /opt/WebSphere/AppServer/installedApps/WC\_Enterprise\_App\_*nome\_istanza*.ear/lib/loader/idresgen.zip
- ▶ **400** /QIBM/UserData/WebASAdv4/WASinstancename/installedApps/WC\_Enterprise\_App\_*nome\_istanza*.ear/lib/loader/idresgen.zip

**Nota:** Se si desidera specificare un proprio file di personalizzazione, è necessario aggiungere la seguente sezione DevTools al file *nome\_istanza.xml*:

- `IDResolverCustomizerFile="myIDResolverCustomizerFile"`

Per impostazione predefinita il file di configurazione di WebSphere Commerce, *nome\_istanza.xml*, non specifica un valore per questo attributo.

ID Resolver utilizza *nomenegoziomaster.xml* e il file DTD corrispondente, *wcs.dtd*. Una volta risolti gli ID, ID Resolver crea il file *formato\_orarionomenegoziomaster.xml* che contiene gli identificativi univoci. Se si verifica un errore durante il processo di risoluzione dell'ID, il pacchetto Loader crea un file *error.xml*, *nomenegoziomaster.error.xml*.

**Nota:** Il processo di pubblicazione salva automaticamente questi file temporanei quando un tentativo di pubblicazione non riesce. Se una pubblicazione riesce, invece, tali file vengono eliminati per impostazione predefinita. E' possibile conservare tali file per motivi di risoluzione dei problemi o per pubblicare e utilizzare un negozio nell'ambiente di prova WebSphere. Per salvare i file temporanei, consultare la sezione relativa ai parametri di Servizi del negozio nella guida in linea di WebSphere Commerce.

I file *formato\_orarionomenegoziomaster.xml* e *nomenegoziomaster.error.xml* risiedono nella seguente directory:

-  unità:\WebSphere\CommerceServer\temp\*nome\_istanza*\tools\devtools
-  unità:\Program Files\WebSphere\CommerceServer\temp\*nome\_istanza*\tools\devtools
-  /usr/WebSphere/CommerceServer/temp/*nome\_istanza*/tools/devtools
-  /opt/WebSphere/CommerceServer/temp/*nome\_istanza*/tools/devtools
-  /opt/WebSphere/CommerceServer/temp/*nome\_istanza*/tools/devtools
-  /QIBM/UserData/WebCommerce/instances/*nome\_istanza*/temp/tools/devtools

#### **Richiama il pacchetto Loader per caricare il file XML principale risolto nel**

**database:** Il pacchetto Loader carica il file *formato\_orarionomenegoziomaster.xml* nel database. Se si verifica un errore durante il processo di caricamento, il pacchetto Loader crea un file *error.xml*, *formato\_orarionomenegoziomaster.error.xml*.

Per ulteriori informazioni sul pacchetto Loader, consultare la Capitolo 27, "Panoramica sul caricamento dei dati" a pagina 229.

Quando Servizi del negozio o la funzione di pubblicazione della riga comandi richiamano il pacchetto Loader, è necessario specificare quale metodo utilizzare. Servizi del negozio può utilizzare uno dei seguenti metodi di Loader:

- SQL import
- Import
- Load

**Nota:** Per impostazione predefinita, Servizi del negozio utilizza il metodo misto. E' possibile specificare il metodo da utilizzare con Servizi del negozio o con il

programma di utilità della riga comandi nel file di configurazione di WebSphere Commerce, *nome\_istanza.xml*, mediante l'attributo LoaderMode dell'elemento DevTools.

- **SQL Import:** questo metodo utilizza Java (Java Database Connectivity) per inserire e aggiornare dati e costituisce il metodo più flessibile ma più lento per importare grandi quantità di dati in un piccolo numero di tabelle. Consente un aggiornamento a livello della colonna. Si consiglia di utilizzare l'importazione SQL.

**Nota:** Il metodo SQL import è il mezzo più sicuro da utilizzare perché non corrompe il database quando i dati non sono validi. Prima di eseguire un caricamento mediante SQL import, i record devono soddisfare le limitazioni dello schema di database. Gli altri metodi di Loader sono più veloci perché i record vengono caricati in massa senza verifiche. Di conseguenza, prima di utilizzare gli altri metodi, è necessario essere certi della correttezza dei dati.

- **Import:** questo metodo utilizza le funzioni di importazione native di DB2 e consente l'aggiornamento a livello della cella con una velocità e una flessibilità medie. Questo metodo non è disponibile con Oracle.
- **Load:** questo metodo utilizza le funzioni native di RDBMS (DB2 Load o SQLLoad) ed è il metodo più rapido per il caricamento di grandi quantità di dati in un piccolo numero di tabelle.

Per ulteriori informazioni sui metodi del comando load, consultare "Comando Load" a pagina 242.

Servizi del negozio e la funzione di pubblicazione della riga comandi devono specificare anche un file di personalizzazione da utilizzare con Loader. Se non si specifica un file di personalizzazione nel file di configurazione di WebSphere Commerce, *nome\_istanza.xml*, il codice di pubblicazione utilizza il file di personalizzazione predefinito: MassLoadCustomizer.

**Nota:** Se si desidera specificare un proprio file di personalizzazione, è necessario aggiungere la seguente sezione DevTools al file *nome\_istanza.xml*:

- `LoaderCustomizerFile="FilePersonalizzazioneLoaderUtente"`

Per impostazione predefinita il file di configurazione di WebSphere Commerce, *nome\_istanza.xml*, non specifica un valore per questo attributo.

## Crea il file parameters.jsp

Il processo di pubblicazione crea il file parameters.jsp. Questo file include tre parametri: storeId, catalogId e langId. Il file index.jsp dei negozi di esempio utilizza questi parametri per avviare il negozio.

Affinché il processo di pubblicazione crei il file parameters.jsp, i dati di catalogo e le risorse Web devono essere pubblicate simultaneamente almeno una volta. Se questi due elementi non vengono pubblicati insieme, Servizi del negozio è in grado di avviare il negozio.

parameters.jsp si trova nella seguente directory:

-  unità:\WebSphere\AppServer\installedApps\WC\_Enterprise\_App\_*nomeistanza*.ear\wcstores.war\dirnegozio\include
-  unità:\Program Files\WebSphere\AppServer\installedApps\WC\_Enterprise\_App\_*nomeistanza*.ear\wcstores.war\dirnegozio\include

-  /usr/WebSphere/AppServer/installedApps/  
WC\_Enterprise\_App\_nomeistanza.ear/wcstores.war/dirnegozio/include
-  /opt/WebSphere/AppServer/installedApps/  
WC\_Enterprise\_App\_nomeistanza.ear/wcstores.war/dirnegozio/include
-  /opt/WebSphere/AppServer/installedApps/  
WC\_Enterprise\_App\_nomeistanza.ear/wcstores.war/dirnegozio/include
-  /QIBM/UserData/WebASAdv4/WASinstancename/installedApps/  
WC\_Enterprise\_App\_nomeistanza.ear/wcstores.war/dirnegozio/include

## Spacchettamento dei file di configurazione del negozio

Gli archivi dei negozi di esempio  ToolTech e NewFashion includono anche i seguenti file:

- tools\_properties.zip
- tools\_xml.zip
- runtime\_xml.zip

Tali file sono registrati in sarinfo.xml file e vengono utilizzati da Servizi del negozio per configurare i negozi. Durante il processo di pubblicazione questi file vengono spacchettati e copiati nelle directory specificate nel file di configurazione di WebSphere Commerce, *nome\_istanza.xml*. Per ulteriori informazioni, fare riferimento alla sezione “Pubblicazione dei file di configurazione e delle risorse facciata del negozio mediante copia su WebSphere Commerce Server” a pagina 290.

**Nota:** Tali file non devono essere modificati, rimossi, copiati in altri negozi o salvati in un'altra directory.

## Richiama il comando per pubblicare contratti e conti commerciali

Alcune delle risorse database del negozio (contratti e conti commerciali) non possono essere caricate dal pacchetto Loader. Pertanto, per poter pubblicare tali risorse su WebSphere Commerce Server, la funzione di pubblicazione richiama i comandi corrispondenti. Tali comandi sono i seguenti:

- AccountImport — Crea un conto commerciale dal file businessaccount.xml nell'archivio negozio.
- ContractImportApprovedVersion — Importa un contratto dal file contract.xml nell'archivio negozio.
- ProductSetPublish — Sincronizza i dati dell'insieme di prodotti nel database con il catalogo prima della creazione di contratti e conti commerciali. Servizi del negozio e il programma di utilità per la pubblicazione della riga comandi richiamano il comando ProductSetPublish che a sua volta richiama i comandi AccountImport e ContractImportApprovedVersion.

Per ulteriori informazioni sulla pubblicazione di contratti e conti commerciali, consultare Capitolo 29, “Pubblicazione dei contratti e dei conti commerciali” a pagina 285.

## Aggiorna i componenti del registro

L'ultima azione del processo di pubblicazione è l'aggiornamento dei componenti del registro. Il programma di utilità per la pubblicazione aggiorna tutti i registri di WebSphere Commerce. Per ulteriori informazioni sui registri, consultare la guida in linea di WebSphere Commerce.

## Gestione degli errori

Se un errore si verifica durante la fase di pubblicazione delle risorse, è possibile visualizzare un messaggio di errore nei log di pubblicazione (vedere “File di log della pubblicazione”) o con la pagina di riepilogo della pubblicazione nei Servizi del negozio.

## Configurazione del pagamento

L'ultimo passo del processo di pubblicazione consiste nel configurare il pagamento. WebSphere Commerce supporta IBM Payment Manager. Se si desidera utilizzare Payment Manager come metodo di elaborazione dei pagamenti, è necessario creare un file XML di pagamenti come descritto in Capitolo 13, “Risorse pagamento” a pagina 119. Se un file XML di pagamenti è incluso nell'archivio negozio da pubblicare, WebSphere Commerce completa la seguente configurazione di pagamento durante la pubblicazione:

- Creazione del il venditore.
- Creazione del conto (solo per cassette non in linea).
- Creazione dei marchi specificati in `paymentinfo.xml` (solo per le cassette non in linea).
- Assegnazione dell'autorità all'utente.

## Gestione degli errori

Se un errore si verifica durante la fase di configurazione dei pagamenti, è possibile visualizzare il messaggio di errore nei log di pubblicazione (vedere “File di log della pubblicazione”).

## File di log della pubblicazione

Qualsiasi errore rilevato durante la fase di pubblicazione delle risorse viene scritto nei seguenti file di log e traccia:

- `messages.txt`: contiene i messaggi di errore della parte del processo di pubblicazione relativa al pacchetto Loader. Quando una pubblicazione non riesce, verificare innanzitutto questo file. I numeri di riga o colonna indicati in questi messaggi di errore si riferiscono ai file `master.xml` temporanei: `nomenegoziomaster.xml`, o `formato_orarionomenegoziomaster.xml`.
- `trace.txt`: contiene informazioni di traccia sulle parti del processo di pubblicazione relative al pacchetto Loader e a ID Resolver. Per impostazione predefinita, `trace.txt` è disabilitato.
- `ecmsg_nomeistanza_dataora.log`: registra tutti i messaggi di errore di WebSphere Commerce Server.
- `wcs.log`: contiene l'emissione di tutte le applicazioni (incluso la pubblicazione) in esecuzione sul server delle applicazioni WebSphere sulla console standard.

I file di log risiedono nella seguente directory:

-  `NT` unità:\WebSphere\CommerceServer\instances\`nomeistanza`\logs
-  `2000` unità:\Program Files\WebSphere\CommerceServer\`nomeistanza`\logs
-  `AIX` /usr/WebSphere/CommerceServer/instances/`nomeistanza`/logs
-  `Solaris` /opt/WebSphere/CommerceServer/instances/`nomeistanza`/logs
-  `Linux` /opt/WebSphere/CommerceServer/instances/`nomeistanza`/logs
-  `400` /QIBM/UserData/WebCommerce/instances/`nomeistanza`/logs

Per configurare i file di log `trace.txt` e `messages.txt` (ovvero per regolare il livello di log o eseguire altre opzioni), modificare il seguente file:

- **NT** unità:\WebSphere\CommerceServer\xml\loader\WCALoggerConfig.xml
- **2000** unità:\Program Files\WebSphere\CommerceServer\xml\loader\WCALoggerConfig.xml
- **AIX** /usr/WebSphere/CommerceServer/xml/loader/WCALoggerConfig.xml
- **Solaris** /opt/WebSphere/CommerceServer/xml/loader/WCALoggerConfig.xml
- **Linux** /opt/WebSphere/CommerceServer/xml/loader/WCALoggerConfig.xml
- **400** /QIBM/UserData/WebCommerce/instances/instancename/xml/WCALoggerConfig.xml

**Nota:** Per ulteriori informazioni sulla configurazione del file WCALoggerConfig.xml, consultare il manuale *WebSphere Commerce Catalog Manager - Guida per l'utente*.



---

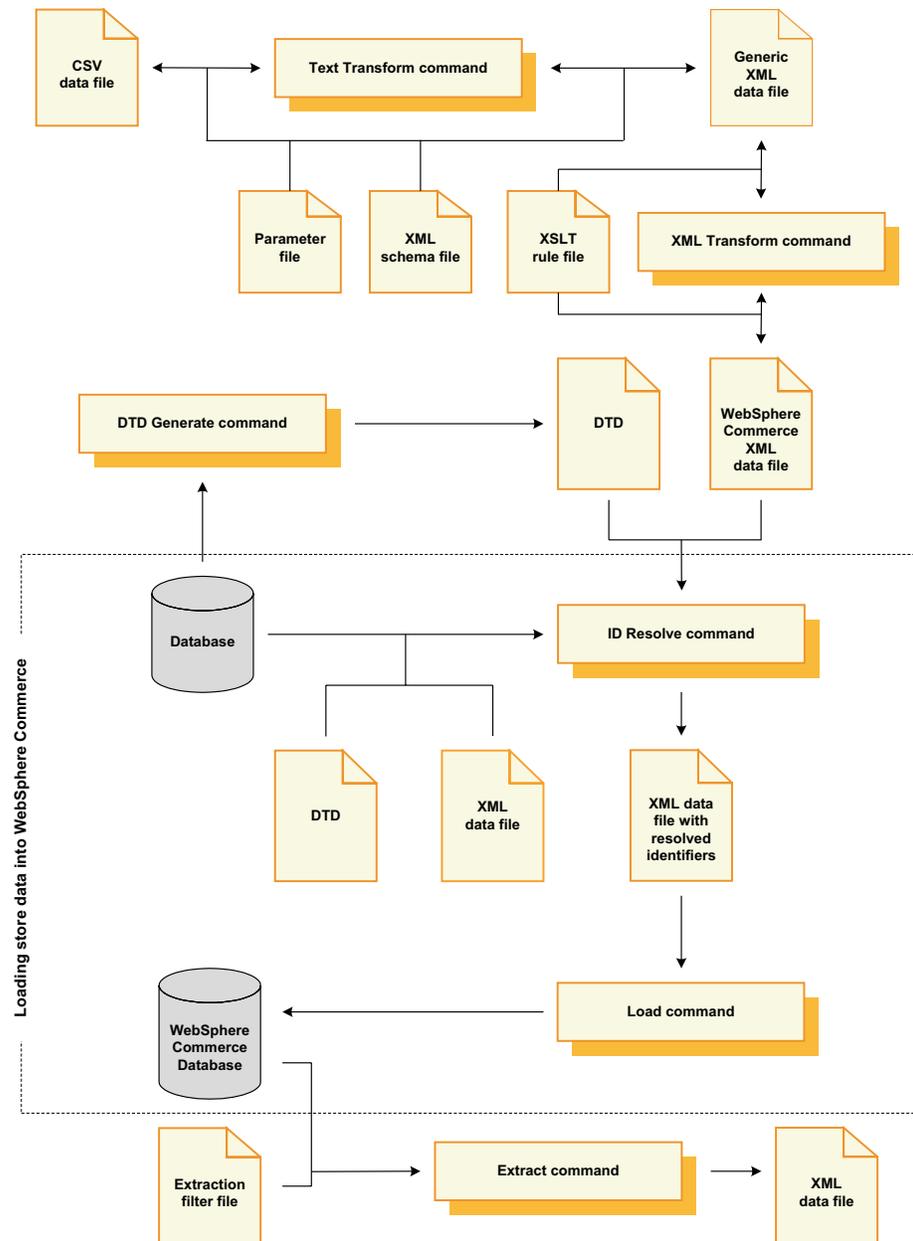
## Capitolo 27. Panoramica sul caricamento dei dati

Una volta creati i dati del negozio, è possibile scegliere di impacchettarli come archivio negozio e pubblicarli mediante Servizi del negozio oppure caricarli direttamente nel database del server WebSphere Commerce utilizzando il pacchetto Loader di WebSphere Commerce Catalog Manager. Per informazioni sul processo di caricamento dei gruppi di risorse database di WebSphere Commerce, fare riferimento a Capitolo 28, "Caricamento dei gruppi di risorse di database di WebSphere Commerce" a pagina 271 e "Caricamento dei gruppi di risorse database" a pagina 279.

Catalog Manager fornisce sei utilità di riga comandi (denominati collettivamente "pacchetto Loader") e tre strumenti di gestione associati che è possibile utilizzare per preparare i dati per il caricamento e caricarli nel negozio. Questi comandi e strumenti utilizzano file di dati XML (Extensible Markup Language) per gestire le informazioni.

## Comprensione del caricamento dati in WebSphere Commerce

La figura di seguito riportata illustra i processi di estrazione, caricamento e preparazione dei dati che è possibile eseguire con il pacchetto Loader.



Tener presente che una linea tratteggiata indica i due processi più comunemente utilizzati per caricare i dati del negozio nel database del server WebSphere Commerce: risoluzione degli identificativi e caricamento dei dati. Questi processi sono descritti in questo capitolo

Per informazioni sulla preparazione dei dati per il caricamento nel database del server WebSphere Commerce, fare riferimento alla Parte 4, "Sviluppo dei dati del negozio" a pagina 37.

Le seguenti due utilità di riga comandi del pacchetto Loader sono comunemente utilizzate per caricare i dati in un database del server WebSphere Commerce:

- **Comando ID Resolve**

Per caricare i dati XML in un database del server WebSphere Commerce mediante il pacchetto Loader, gli elementi XML devono essere associati direttamente allo schema di destinazione del server WebSphere Commerce. Tutti gli elementi XML che hanno attributi che corrispondono a chiavi univoche o primarie nello schema di database devono avere identificativi univoci e tutte le colonne dello schema di database che non possono essere annullate devono avere attributi corrispondenti definiti con valori non nulli. ID Resolver può generare identificativi univoci per attributi di chiavi univoche o primarie di elementi XML di qualificazione.

**Nota:** Come indicato in questo documento, un identificativo è un valore presente in una singola colonna di una tabella di database che fornisce a ciascun a riga un'identità univoca. Se si utilizza ID Resolver per generare identificativi, si ottiene un valore di base dalla tabella base KEYS o SUBKEYS che viene incrementato in modo sequenziale in modo da risolvere un identificativo per ciascuna riga della tabella di database .

Per informazioni su questo comando, fare riferimento a "Comando ID Resolve" a pagina 234, "Uso dei comandi e degli script del pacchetto Loader" a pagina 261 e "Esempi di risoluzione degli identificativi" a pagina 262.

- **Comando Load**

Il pacchetto Loader utilizza file XML validi e in formato corretto come immissione per caricare i dati nel database. Gli elementi del documento XML vengono associati ai nomi della tabella nel database e gli attributi dell'elemento vengono associati alle colonne.

**Nota:** Fare riferimento alle direttive XML WC3 (World Wide Web Consortium (W3C) per una descrizione dei limiti sulla correttezza dei formati e sulla validità.

Per informazioni su questo comando, fare riferimento a "Comando Load" a pagina 242, "Uso dei comandi e degli script del pacchetto Loader" a pagina 261 e "Esempio di caricamento dei dati" a pagina 269.

Questi comandi sono descritti in questo capitolo

Per gestire i dati, è possibile utilizzare anche le seguenti utilità di riga comandi del pacchetto Loader:

- **Comando DTD Generate**

DTD Generator genera una DTD (Document Type Definition) che descrive le tabelle e le colonne del database di destinazione in cui devono essere caricati i dati XML. DTD Generator può generare anche uno schema XML per il database. DTD Generator può creare una DTD basata sullo schema di database WebSphere Commerce. Se si utilizzano le DTD fornite con gli archivi negozio di esempio e non si modifica lo schema di database, in genere non è necessario generare una DTD con DTD Generator.

Per ulteriori informazioni, fare riferimento a "Comando DTD Generate" a pagina 250.

- **Comando Extract**

Il programma Extractor utilizza un'interrogazione rivolta a un database per estrarre sottoinsiemi selezionati di dati dal database in un documento XML.

E' possibile utilizzare questo comando per estrarre i dati dal database in formato XML.

Per ulteriori informazioni, fare riferimento a "Comando Extract" a pagina 253.

- **Comando Text Transform**

Text Transformer converte i dati da un formato variabile delimitato da caratteri a un formato dati XML.

Se i dati non possono essere estratti direttamente da un database in formato XML, ad esempio, è possibile salvarli in un formato variabile delimitato da caratteri e quindi utilizzare questo comando per convertirlo in formato XML.

Per ulteriori informazioni, fare riferimento a “Comando Text Transform” a pagina 255.

- **Comando XML Transform**

XML Transformer converte i dati di un documento XML in un formato XML alternativo. Esso utilizza XSL (Extensible Stylesheet Language) per definire le regole di corrispondenza per la trasformazione.

È possibile utilizzare questo comando per convertire i dati XML in un formato che corrisponda direttamente allo schema del database WebSphere Commerce di destinazione nel quale si desidera caricare i dati.

Per ulteriori informazioni, fare riferimento a “Comando XML Transform” a pagina 256.

Questi comandi non vengono descritti in dettaglio in questo capitolo. Per informazioni dettagliate su questi comandi, fare riferimento alla versione più recente di *IBM WebSphere Commerce 5.4 Catalog Manager - Guida per l'utente*.

WebSphere Commerce Catalog Manager comprende anche i due strumenti di seguito riportati come guida per le funzioni di gestione dati:

- **Strumento Text Transformation**

Text Transformation consente di eseguire una conversione di dati da un formato variabile delimitato da caratteri in formato XML mediante il comando Text Transform.

- **XSL editor**

XSL editor fornisce un'interfaccia visiva per la modifica dei file XSL che possono essere utilizzati da XML Transformer. Mediante XSL Editor, è possibile stabilire l'associazione tra un elemento della DTD di origine e un elemento di quella di destinazione durante la definizione delle regole di corrispondenza per la conversione dei dati tra formati XML.

- **Web editor**

Web editor consente di creare, modificare ed eliminare i dati di un database con un browser Web.

Questi strumenti non vengono descritti in dettaglio in questo capitolo. Per informazioni dettagliate, fare riferimento alla versione più recente del manuale *IBM WebSphere Commerce 5.4 Catalog Manager - Guida per l'utente*.

## Comandi del pacchetto Loader per il caricamento dei dati del negozio

### Comando ID Resolve

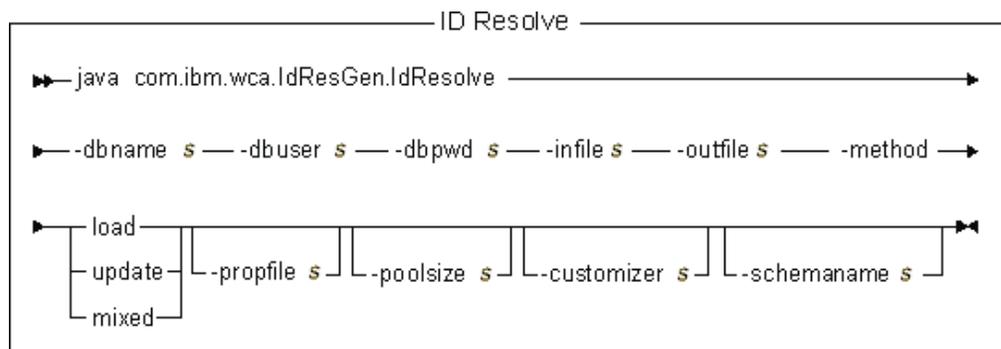
Questo comando genera gli identificativi per gli elementi di dati XML che richiedono il caricamento in un database. Se i dati XML di origine forniscono già gli identificativi univoci necessari, non è necessario eseguire ID Resolver.

Lo schema di database WebSphere Commerce definisce le chiavi primarie ed esterne all'interno delle tabelle utilizzate per rappresentare diverse relazioni tra le tabelle. Per questo motivo, gli elementi XML di WebSphere Commerce devono contenere attributi corrispondenti con identificativi univoci. Nel database del server WebSphere Commerce, le tabelle con identificativi che devono essere risolti sono quelle definite nelle tabelle KEYS e SUBKEYS. Queste tabelle sono chiamate *tabelle primarie* in WebSphere Commerce. Per ulteriori informazioni sulle tabelle KEYS e SUBKEYS, consultare la guida in linea di WebSphere Commerce.

**Nota:** Se occorre risolvere gli identificativi di una tabella non definita nella tabella KEYS o SUBKEYS, aggiungere la tabella alla tabella SUBKEYS prima di eseguire ID Resolver.

Poiché gli attributi e gli elementi XML di WebSphere Commerce sono stati progettati per essere trasferibili su più database e istanze di database, gli identificativi sono di solito rappresentati con alias interni. Prima di poter caricare i dati in un database del server WebSphere Commerce, è necessario risolvere gli alias in identificativi numerici validi. Per ulteriori informazioni, consultare il Appendice B, "Creazione di dati" a pagina 319.

Windows AIX Solaris Linux



#### Note:

1. Il grafico sopra riportato rappresenta principalmente un riferimento per i parametri del comando. Lo script o il file di comando fornito per questo comando e elencato in "Uso dei comandi e degli script del pacchetto Loader" a pagina 261 funge da wrapper per il comando Java effettivo e accetta gli stessi parametri; pertanto, si consiglia di utilizzare lo script o il file di comando invece di richiamare direttamente il comando Java.
2. I nomi file specificati come parametri per questo comando possono essere preceduti da percorsi relativi o assoluti.

## Valori dei parametri:

### **-dbname**

Il nome del database di destinazione

### **-dbuser**

Il nome dell'utente che si collega al database

### **-dbpwd**

La password dell'utente che si collega al database

**-infile** Il nome del documento XML di immissione che contiene i record della tabella

### **-outfile**

Il nome del file XML di emissione da produrre; tale file deve essere utilizzato come immissione per il pacchetto Loader

### **-method**

Metodo da utilizzare nell'elaborazione del file di immissione.

- Utilizzare il metodo load per elaborare il file di immissione se *tutti* i record del file *non esistono* nel database.
- Utilizzare il metodo update per elaborare il file di immissione se *tutti* i record del file *esistono* nel database.
- Utilizzare il metodo misto per elaborare il file di immissione se *solo alcuni* record del file *esistono* nel database.

Il metodo predefinito è il metodo load.

### **-profile**

Il file di testo contenente le proprietà Java sotto forma di coppie nome=valore. Questa proprietà imposta il modo in cui ID Resolver risolve gli identificativi. Viene utilizzato per descrivere quali colonne di una voce primaria devono essere utilizzate come ricerca per le tabelle che richiedono l'identificativo di una riga primaria. Questo file definisce i nomi colonna per la ricerca di identificativi di chiavi straniere e il predicato di selezione per le interrogazioni delle tabelle principali (ad esempio CATEGORY e PRODUCT). E' possibile omettere le voci in questo file per le tabelle che hanno un indice univoco definito che non include l'identificativo. Questo parametro è facoltativo. `IdResolveKeys.properties` è il file predefinito. Questo file di proprietà può essere specificato come indicato in uno dei seguenti esempi:

```
-profile d:\WebSphere\CommerceServer\prop\idresprop.properties
```

```
-profile d:\WebSphere\CommerceServer\prop\idresprop
```

Se il file esiste nella directory corrente, è possibile specificare lo stesso file come illustrato nel seguente esempio:

```
-profile idresprop.properties
```

Se questo file si trova in una directory specificata nella variabile di ambiente del sistema classpath, lo stesso file può essere specificato come illustrato nel seguente esempio:

```
-profile idresprop
```

Per ulteriori informazioni sulla creazione e la specificazione di un nuovo file properties con ID Resolver, fare riferimento alla versione più recente del manuale *IBM WebSphere Commerce 5.4 Catalog Manager - Guida per l'utente*.

#### -poolsize

Il numero di identificativi da riservare. Questo parametro è facoltativo. Il numero predefinito è 50.

#### -customizer

Il nome del file properties di personalizzazione da utilizzare. Questo parametro è facoltativo. Il file properties di personalizzazione imposta il modo in cui funziona ID Resolver. `DB2ConnectionCustomizer.properties` è il file predefinito. Il file properties del cliente può essere specificato come illustrato in uno dei seguenti esempi:

```
-customizer d:\WebSphere\CommerceServer\prop\idres.properties
-customizer d:\WebSphere\CommerceServer\prop\idres
```

Se il file esiste nella directory corrente, è possibile specificare lo stesso file come illustrato nel seguente esempio:

```
-customizer idres.properties
```

Se questo file si trova in una directory specificata nella variabile di ambiente del sistema classpath, lo stesso file può essere specificato come illustrato nel seguente esempio:

```
-customizer idres
```

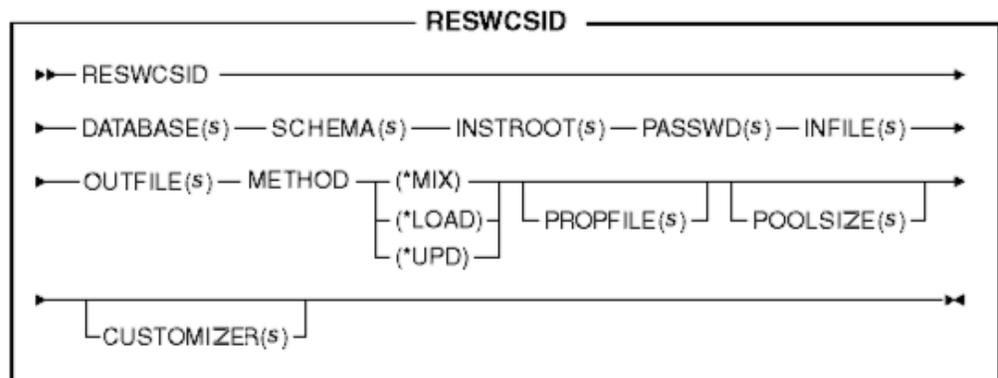
Per ulteriori informazioni sulla creazione e la specificazione di un nuovo file properties di personalizzazione, fare riferimento alla versione più recente del manuale *IBM WebSphere Commerce 5.4 Catalog Manager - Guida per l'utente*.

#### -schemaname

Il nome dello schema del database di destinazione. Questo parametro è facoltativo.

Se questo parametro non è specificato durante l'esecuzione del comando, il comando cerca un'altra coppia nome=valore nel file properties di personalizzazione che specifica il valore di SchemaName. Se la coppia è presente nel file delle proprietà, il comando utilizza il valore specificato. Se per questo parametro non esiste specifica di file delle proprietà o di riga comandi, il comando utilizza per impostazione predefinita il nome schema della tabella nel database.

▶ 400



**Nota:** I nomi file specificati come parametri per questo comando possono essere preceduti da percorsi relativi o assoluti.

## Valori dei parametri:

### DATABASE

Il nome del database di destinazione, come visualizzato nella directory del database relazionale

### SCHEMA

Il nome dello schema del database di destinazione, uguale al nome dell'istanza

### INSTROOT

Il nome completo del percorso root dell'istanza di WebSphere Commerce, ad esempio `/QIBM/UserData/WebCommerce/instances/nome_istanza`

### PASSWD

Password dell'istanza di WebSphere Commerce

### INFILE

Il nome del documento XML di immissione che contiene i record della tabella

### OUTFILE

Il nome del file XML di emissione da produrre; tale file deve essere utilizzato come immissione per il pacchetto Loader

### METHOD

Metodo da utilizzare nell'elaborazione del file di immissione.

- Utilizzare il metodo load (\*LOAD) per elaborare il file di immissione se *tutti i* record del file *non esistono* nel database.
- Utilizzare il metodo update (\*UPD) per elaborare il file di immissione se *tutti i* record del file *esistono* nel database.
- Utilizzare il metodo misto (\*MIX) per elaborare il file di immissione se *solo alcuni* record del file *esistono* nel database.

### PROFILE

Il file di testo contenente le proprietà Java sotto forma di coppie nome=valore. Questa proprietà imposta il modo in cui ID Resolver risolve gli identificativi. Viene utilizzato per descrivere quali colonne di una voce primaria devono essere utilizzate come ricerca per le tabelle che richiedono l'identificativo di una riga primaria. Questo file definisce i nomi colonna per la ricerca di identificativi di chiavi straniere e il predicato di selezione per le interrogazioni delle tabelle principali (ad esempio CATEGORY e PRODUCT). E' possibile omettere le voci in questo file per le tabelle che hanno un indice univoco definito che non include l'identificativo. Questo parametro è facoltativo. `IdResolveKeys.properties` è il file predefinito. Questo file di proprietà può essere specificato come indicato in uno dei seguenti esempi:

```
PROFILE(/wc/prop/idresprop.properties)
```

```
PROFILE(/wc/prop/idresprop)
```

Se il file esiste nella directory corrente, è possibile specificare lo stesso file come illustrato nel seguente esempio:

```
PROFILE(idresprop.properties)
```

Se questo file si trova in una directory specificata nella variabile di ambiente del sistema classpath, lo stesso file può essere specificato come illustrato nel seguente esempio:

```
PROFILE(idresprop)
```

Per ulteriori informazioni sulla creazione e la specificazione di un nuovo file properties con ID Resolver, fare riferimento alla versione più recente del manuale *IBM WebSphere Commerce 5.4 Catalog Manager - Guida per l'utente*.

**POOLSIZE**

Il numero di identificativi da riservare. Questo parametro è facoltativo. Il numero predefinito è 50.

## CUSTOMIZER

Il nome del file properties di personalizzazione da utilizzare. Questo parametro è facoltativo. Il file properties di personalizzazione imposta il modo in cui funziona ID Resolver. Il file predefinito è `ISeries_RESWCSID_Customizer.properties`. Il file properties del cliente può essere specificato come illustrato in uno dei seguenti esempi:

```
CUSTOMIZER(/wc/prop/idres.properties)
```

```
CUSTOMIZER(/wc/prop/idres)
```

Se il file esiste nella directory corrente, è possibile specificare lo stesso file come illustrato nel seguente esempio:

```
CUSTOMIZER(idres.properties)
```

Se questo file si trova in una directory specificata nella variabile di ambiente del sistema classpath, lo stesso file può essere specificato come illustrato nel seguente esempio:

```
CUSTOMIZER(idres)
```

Per ulteriori informazioni sulla creazione e la specificazione di un nuovo file properties di personalizzazione, fare riferimento alla versione più recente del manuale *IBM WebSphere Commerce 5.4 Catalog Manager - Guida per l'utente*.

### Tecniche di risoluzione:

ID Resolver risolve gli identificativi mediante una combinazione di due o tre delle seguenti tecniche, a seconda che sia utilizzato o meno un file di proprietà.

- **Risoluzione alias interno**

Quando si utilizza la risoluzione dell'ID dell'alias interno, un alias sostituisce la chiave univoca (identificativo) nel documento XML di origine. L'alias viene quindi utilizzato altrove nel file XML per indicare tale elemento.

Gli alias interni devono essere utilizzati in maniera coerente nel file XML. Ad esempio, se un ID rubrica, `ADDRBOOK_ID`, ha come alias `@addrbook_1`, tutti i riferimenti della chiave esterna a quell'ID nel file devono utilizzare `@addrbook_1`.

Gli alias sono transitori. Non vengono salvati e non possono essere utilizzati in un file XML separato senza introdurre di nuovo l'alias. Durante la pubblicazione in Servizi del negozio, comunque, i file XML vengono concatenati in modo che la risoluzione possa verificarsi su tutti i dati.

- **Risoluzione indice univoco**

ID Resolver può inoltre analizzare lo schema del database per determinare se esiste un indice univoco che soddisfa i suoi requisiti. ID Resolver cerca un indice univoco soltanto quando non esistono voci nel file properties per la tabella da analizzare o quando non esiste il file properties. Se queste condizioni sono vere, viene eseguita una verifica dell'indice univoco. L'indice univoco viene considerato valido se esiste e non include la chiave primaria per la tabella.

- **Specifica file properties**

ID Resolver consente di utilizzare un file properties Java alternativo per descrivere le colonne di una voce primaria da utilizzare come ricerca delle tabelle che richiedono un identificativo della riga principale.

Gli archivi negozio di esempio forniti con WebSphere Commerce utilizzano alias interni nei file XML. Ciò consente agli archivi negozio di essere trasferibili tra i database. Sebbene le tecniche di specificazione dei file properties e degli indici

univoci consentono anche la trasferibilità tra database, un utente può modificare l'identità delle colonne univoche in qualsiasi momento causando così problemi quando queste tecniche vengono successivamente utilizzate per la risoluzione degli ID. Se un utente modifica una colonna univoca, ad esempio, il nome colonna deve poi essere modificato nella definizione del file delle proprietà. Con la tecnica dell'alias interno, invece, una modifica apportata nel database non richiede la modifica del file properties o XML. Durante la pubblicazione in Servizi del negozio o mediante il pacchetto Loader, ID Resolver sostituisce l'alias con un valore univoco. Una volta caricati i dati, gli alias diventano trasparenti per l'utente. Per ulteriori informazioni, consultare l'Appendice B, "Creazione di dati" a pagina 319.

ID Resolver utilizza il seguente processo:

- Se i dati di immissione XML hanno un elemento di una tabella primaria che ha già un identificativo a codice hardware (ad esempio "12345"), ID Resolver non crea un nuovo identificativo per quell'elemento.
- Se i dati di immissione XML hanno un elemento di una tabella primaria senza identificativo, ID Resolver verifica se nel database è già presente una riga per quell'elemento.

La ricerca dell'elemento nel database richiede che le altre colonne dell'elemento siano utilizzate per formare una chiave univoca. Queste colonne possono essere specificate nel file properties; oppure è possibile fare in modo che ID Resolver stabilisca le colonne da utilizzare.

- Se un file properties viene utilizzato e contiene una voce per la tabella da analizzare, ID Resolver utilizza le colonne specificate nel file per formare la chiave univoca.
- Se non viene utilizzato alcun file properties o se esso non contiene una voce per la tabella da analizzare, ID Resolver utilizza la risoluzione indice univoco. Questo tipo di risoluzione utilizza un qualsiasi indice univoco specificato su una tabella come mezzo per la localizzazione dell'identificativo. Ad esempio, MEMBER\_ID oltre a IDENTIFIER è un indice univoco sulla tabella CATALOG e perciò può essere utilizzato come un punto di risoluzione nella chiave esterna CATALOG\_ID della tabella CATALOGDSC.

L'elemento esiste già nel database se è presente una riga con la stessa chiave univoca; altrimenti viene considerato un nuovo dato.

- Se l'elemento esiste già come riga nel database, l'identificativo viene richiamato e salvato per poter essere utilizzato in un momento successivo. Altrimenti, viene generato un altro identificativo generato da ID Resolver mediante un valore disponibile nella tabella KEYS o SUBKEYS.
- Se è stato specificato un alias interno per l'elemento ("@store\_id\_1" ad esempio) nel documento XML, tale alias è associato all'identificativo affinché quest'ultimo possa essere cercato in un secondo momento con lo stesso alias interno.
- Gli elementi successivi del documento XML che richiedono di fare riferimento a un elemento della tabella primaria utilizzano l'alias interno se l'elemento della tabella primaria ne ha uno (ad esempio "@store\_id\_1") o, in caso contrario, i valori delle colonne di ricerca (ad esempio "@WC2001@100"). In entrambi i casi, il valore specificato viene utilizzato per ricercare l'identificativo effettivo con il quale viene poi sostituito.
- Quando il documento di emissione XML viene prodotto, tutti gli elementi della tabella primaria hanno identificativi effettivi e tutti gli elementi che si riferiscono a tali elementi utilizzano gli identificativi effettivi, non gli alias interni o i valori delle colonne di ricerca. Questo è il documento XML completamente risolto.

### Metodi per il comando ID Resolve:

Il comando ID Resolve consente di scegliere il metodo load, update o mixed per l'elaborazione del file di immissione.

#### Metodo load:

Il metodo load per ID Resolver viene utilizzato per creare nuovi identificativi per tutti i nuovi record caricati nel database.

**Nota:** Se viene specificato il metodo load per ID Resolver, i record nel file di immissione devono essere già presenti nel database. Se il metodo viene utilizzato con ID Resolver e un record del file XML di origine già esiste nel database di destinazione, durante il caricamento dei dati il programma Loader genera un errore. ID Resolver assegna una nuova chiave primaria al record nel file XML durante la risoluzione dell'ID, ma quando i dati vengono caricati nel database, viene generato un errore. Il programma Loader non si arresta al momento di elaborare il record duplicato, ma riporta un errore e il record duplicato non viene caricato nel database.

L'esempio di seguito riportato viene utilizzato per generare identificativi per elementi dati nuovi per il database:

-   

```
idresgen -dbname db -dbuser user -dbpwd pwd -infile input.xml
-outfile output.xml -method load -customizer customizer -schemaname wcsadmin
```
-     

```
./idresgen.sh -dbname db -dbuser user -dbpwd pwd -infile input.xml
-outfile output.xml -method load -customizer customizer -schemaname wcsadmin
```
-   

```
QWEBCOMM/RESWC SID DATABASE(DATABASE_NAME) SCHEMA(WCSADMIN)
INSTROOT(/QIBM/UserData/WebCommerce/instances/mser)
PASSWD(mypassword) INFILE(input.xml) OUTFILE(output.xml)
METHOD(*LOAD)
```

**Nota:** Per la posizione dello script o comando ID Resolve appropriato, fare riferimento a "Uso dei comandi e degli script del pacchetto Loader" a pagina 261.

#### Metodo update:

Se viene specificato il metodo update per ID Resolver, i record nel file di immissione devono essere già presenti nel database. L'ID Resolver localizza gli identificativi nel database, come descritto a pagina 240. Se un record non esiste nel database, ID Resolver non è in grado di risolvere l'identificativo per questo record e indica che si è verificato un errore. Il seguente esempio viene utilizzato per localizzare identificativi per elementi di dati già presenti nel database:

-   

```
idresgen -dbname db -dbuser user -dbpwd pwd -infile input.xml
-outfile output.xml -method update -customizer customizer -schemaname wcsadmin
```
-     

```
./idresgen.sh -dbname db -dbuser user -dbpwd pwd -infile input.xml
-outfile output.xml -method update -customizer customizer -schemaname wcsadmin
```
-   

```
QWEBCOMM/RESWC SID DATABASE(DATABASE_NAME) SCHEMA(WCSADMIN)
INSTROOT(/QIBM/UserData/WebCommerce/instances/mser) PASSWD(mypassword)
INFILE(input.xml) OUTFILE(output.xml) METHOD(*UPD)
```

**Nota:** Per la posizione dello script o comando ID Resolve appropriato, fare riferimento a “Uso dei comandi e degli script del pacchetto Loader” a pagina 261.

*Metodo misto:*

Se il file dei dati di immissione contiene record già presenti nel database e altri record completamente nuovi, ID Resolver deve essere eseguito utilizzando il metodo mixed. Con questo metodo, ID Resolver crea nuovi identificativi per i record soltanto se non esistono nel database. Altrimenti, nel database vengono rilevati gli identificativi già esistenti. Il seguente esempio viene utilizzato per creare identificativi per i dati nuovi e per localizzare gli identificativi per elementi di dati già presenti nel database:

- 

```
idresgen -dbname db -dbuser user -dbpwd pwd -infile input.xml
-outfile output.xml -method mixed -customizer customizer -schemaname wcsadmin
```
-   

```
./idresgen.sh -dbname db -dbuser user -dbpwd pwd -infile input.xml
-outfile output.xml -method mixed -customizer customizer -schemaname wcsadmin
```
- 

```
QWBCOMM/RESWCSID DATABASE(DATABASE_NAME) SCHEMA(WCSADMIN)
INSTROOT(/QIBM/UserData/WebCommerce/instances/mser) PASSWD(mypassword)
INFILE(input.xml) OUTFILE(output.xml) METHOD(*MIX)
```

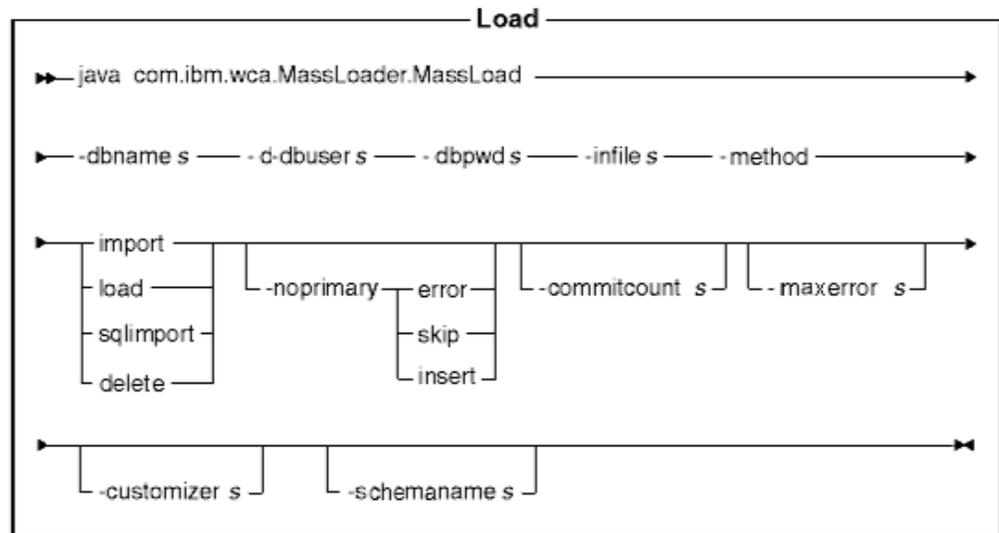
**Note:**

1. Per la posizione dello script o comando ID Resolve appropriato, fare riferimento a “Uso dei comandi e degli script del pacchetto Loader” a pagina 261.
2. Il metodo misto è consigliato per Servizi del negozio.

Per informazioni dettagliate sull'impostazione e la personalizzazione dei file utilizzati per eseguire questo comando, fare riferimento alla versione più recente del manuale *IBM WebSphere Commerce 5.4 Catalog Manager - Guida per l'utente*.

## **Comando Load**

Questo comando carica un file XML di immissione in un database di destinazione.



**Note:**

1. Il grafico sopra riportato rappresenta principalmente un riferimento per i parametri del comando. Lo script o il file di comando fornito per questo comando ed elencato in "Uso dei comandi e degli script del pacchetto Loader" a pagina 261 funge da wrapper per il comando Java effettivo e accetta gli stessi parametri; pertanto, si consiglia di utilizzare lo script o il file di comando invece di richiamare direttamente il comando Java.
2. I nomi file specificati come parametri per questo comando possono essere preceduti da percorsi relativi o assoluti.

**Valori dei parametri:**

**-dbname**

Il nome del database di destinazione

**-dbuser**

Il nome dell'utente che si collega al database

**-dbpwd**

La password dell'utente che si collega al database

**-infile** Il nome del file XML di immissione

**-method**

La modalità di operazione per il pacchetto Loader da utilizzare durante l'inserimento dei dati nel database

- Il metodo load utilizza la funzione di caricamento nativo dal fornitore del database. E' possibile utilizzare il metodo load per i database Oracle locale e remoto, ma il metodo load può essere utilizzato solo per i database DB2 locali.
- Sebbene il metodo import possa essere utilizzato per caricare i dati nei database locali e remoti, viene di norma utilizzato per caricarli nei database DB2 remoti. Questo metodo utilizza l'opzione import o update se disponibili dal fornitore del database. Se viene specificato questo metodo per un database in cui l'opzione import o update non è disponibile, ad esempio Oracle, le istruzioni SQL che utilizzano JDBC consentono di aggiornare il database.

- Il metodo SQL import (sqlimport) può essere utilizzato sia con i database locali che con quelli remoti.
- Il metodo delete elimina i dati dal database.

**-noprimary**

L'operazione che Loader deve eseguire nel caso in cui la chiave primaria risulta mancante per un record in un file di immissione

- L'opzione error indica la chiave principale mancante come un errore per cui il programma deve terminare.
- L'opzione skip salta i record nel file di immissione che non dispongono di una chiave primaria.
- L'opzione insert cerca di inserire o eliminare i dati.

Questo parametro è facoltativo. L'opzione predefinita è error.

**-commitcount**

Il numero dei record elaborati prima che si verifichi il commit del database durante l'utilizzo della modalità di impostazione SQL dell'operazione.

Questo parametro è facoltativo. Il valore predefinito è 1.

**-maxerror**

Il numero di errori dopo il quale il pacchetto Loader termina la modalità di importazione SQL. Questo parametro è facoltativo.

**-customizer**

Il nome del file properties di personalizzazione da utilizzare. Questo parametro è facoltativo. Il file properties di personalizzazione imposta il modo in cui funziona Loader. `MassLoadCustomizer.properties` è il file predefinito. Il file properties del cliente può essere specificato come illustrato nel seguente esempio:

```
-customizer d:\WebSphere\CommerceServer\prop\ml.properties
```

Se questo file si trova in una directory specificata nella variabile di ambiente del sistema `classpath`, lo stesso file può essere specificato come illustrato nel seguente esempio:

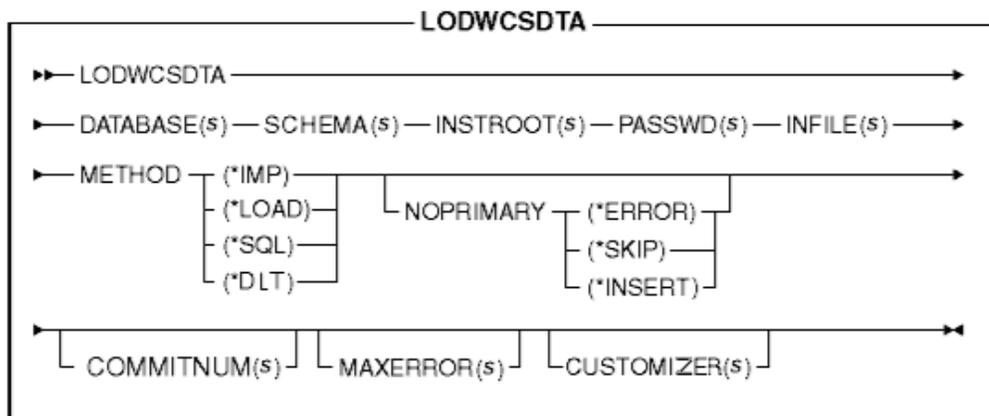
```
-customizer ml
```

Per ulteriori informazioni sulla creazione e la specificazione di un nuovo file properties di personalizzazione, fare riferimento alla versione più recente del manuale *IBM WebSphere Commerce 5.4 Catalog Manager - Guida per l'utente*.

**-schemaname**

Il nome dello schema del database di destinazione. Questo parametro è facoltativo.

Se questo parametro non è specificato durante l'esecuzione del comando, il comando cerca un'altra coppia nome=valore nel file properties di personalizzazione che specifica il valore di `SchemaName`. Se la coppia è presente nel file delle proprietà, il comando utilizza il valore specificato. Se per questo parametro non esiste specifica di file delle proprietà o di riga comandi, il comando utilizza per impostazione predefinita il nome schema della tabella nel database.



**Nota:** I nomi file specificati come parametri per questo comando possono essere preceduti da percorsi relativi o assoluti.

**Valori dei parametri:**

**DATABASE**

Il nome del database di destinazione come visualizzato nella directory del database relazionale

**SCHEMA**

Il nome dello schema del database di destinazione, uguale al nome dell'istanza

**INSTROOT**

Il nome completo del percorso root dell'istanza di WebSphere Commerce, ad esempio /QIBM/UserData/WebCommerce/instances/nome\_istanza

**PASSWD**

Password dell'istanza di WebSphere Commerce

**INFILE**

Il nome del file XML di immissione

**METHOD**

La modalità di operazione per il pacchetto Loader da utilizzare durante l'inserimento dei dati nel database

- Il metodo load (\*LOAD) utilizza la funzione di caricamento nativa del fornitore del database. E' possibile utilizzare il metodo load (\*LOAD) per i database Oracle locale e remoto, ma solo per i database DB2 locali.
- Sebbene il metodo import (\*IMP) possa essere utilizzato per caricare i dati nei database locali e remoti, viene di norma utilizzato per caricarli nei database DB2 remoti. Questo metodo utilizza l'opzione import o update se disponibili dal fornitore del database. Se l'opzione import o update non è disponibile, le istruzioni SQL che utilizzano JDBC consentono di aggiornare il database.
- Il metodo SQL import (\*SQL) può essere utilizzato sia con i database locali che con quelli remoti.
- Il metodo delete (\*DLT) elimina i dati dal database.

**NOPRIMARY**

L'operazione il Loader deve eseguire nel caso in cui la chiave primaria risulta mancante per un record in un file di immissione

- L'opzione error (\*ERROR) indica la chiave principale mancante come un errore per cui il programma deve terminare.
- L'opzione skip (\*SKIP) salta i record nel file di immissione che non hanno una chiave primaria.
- L'opzione insert (\*INSERT) tenta di elaborare (inserire o eliminare) i dati.

Questo parametro è facoltativo. L'opzione predefinita è error.

#### COMMITNUM

Il numero dei record elaborati prima che si verifichi il commit del database durante l'utilizzo della modalità di importazione SQL dell'operazione. Questo parametro è facoltativo. Il valore predefinito è 1.

#### MAXERROR

Il numero di errori dopo il quale il pacchetto Loader termina la modalità di importazione SQL. Questo parametro è facoltativo.

#### CUSTOMIZER

Il nome del file properties di personalizzazione da utilizzare. Questo parametro è facoltativo. Il file properties di personalizzazione imposta il modo in cui funziona ID Resolver. Il file predefinito è `ISeries_LODWCSDTA_Customizer.properties`. Il file properties del cliente può essere specificato come illustrato nel seguente esempio:

```
CUSTOMIZER(/wc/prop/ml.properties)
```

Se questo file si trova in una directory specificata nella variabile di ambiente del sistema classpath, lo stesso file può essere specificato come illustrato nel seguente esempio:

```
CUSTOMIZER(ml)
```

Per ulteriori informazioni sulla creazione e la specificazione di un nuovo file properties di personalizzazione, fare riferimento alla versione più recente del manuale *IBM WebSphere Commerce 5.4 Catalog Manager - Guida per l'utente*.

#### Metodi per il comando Load:

Prima di caricare i dati, è necessario stabilire quale metodo di elaborazione è in grado di produrre i migliori risultati.

*Metodo load:*

Considerare questo metodo in una delle seguenti situazioni:

- I dati di origine sono puliti e il database non contiene dati

**Nota:** La pulizia dei dati non implica alcuna violazione delle limitazioni delle tabelle in cui i dati vengono caricati.

- I dati di origine solo puliti e il database non contiene dati da caricare
- I dati di origine solo puliti, una o più tabelle di destinazione non contiene chiavi primarie e il database non contiene i dati da caricare
- Il database è un database DB2 locale
- Il database è un database Oracle remoto o locale
- Durante il caricamento, al database non accedono utenti o applicazioni

**400** Con il metodo load, i dati vengono caricati nel database. Se i dati già esistono, il comando non viene eseguito a causa di un errore nella chiave duplicata e pertanto viene visualizzato un messaggio di errore di duplicazione.

L'utilizzo del metodo load prevede alcune limitazioni:

- Con il metodo load non è possibile inserire o aggiornare dati nei campi di dati in bit.
-  Con il metodo load, vengono inseriti nel database soltanto record nuovi; i record esistenti non vengono aggiornati.
-  E' possibile utilizzare il metodo load solo per i database DB2 locali, non per quelli remoti.

*Metodo import:*

    Con il metodo import per DB2, i dati vengono caricati nel database. Se i dati sono già presenti non vengono eliminati, ma aggiornati con i nuovi valori. Prendere in considerazione questo metodo nelle seguenti situazioni:

- Il sistema di gestione del database è DB2
- Non si sa se i dati sono puliti
- E' necessario aggiornare insieme estesi di dati omogenei a livello di colonna
- Tutte le tabelle in cui vengono importati i dati hanno chiavi primarie

 Anche con il metodo import i dati vengono caricati nel database. Se i dati sono già presenti, questi non verranno eliminati, ma verranno aggiornati con i nuovi valori. Prendere in considerazione questo metodo nelle seguenti situazioni:

- Non si sa se i dati sono puliti
- I dati sono già presenti nel database
- Tutte le tabelle in cui vengono importati i dati hanno chiavi primarie

L'utilizzo del metodo import prevede alcune limitazioni:

- Il sistema di gestione del database deve essere DB2.
- Con il metodo import non è possibile inserire o aggiornare dati nei campi di dati in bit.
- Con il metodo import, Loader inserisce o aggiorna soltanto le tabelle con chiavi principali definite; tale metodo non consente di inserire o aggiornare dati in tabelle senza chiavi principali. Se soltanto il record di immissione dispone di valori primari per le colonne, il record viene rifiutato.

*Metodo import SQL:*

Con il metodo import SQL, le istruzioni JDBC o SQL vengono utilizzate per aggiornare o inserire i dati nel database. I dati vengono inseriti se non sono già presenti e i dati esistenti vengono aggiornati. Prendere in considerazione questo metodo nelle seguenti situazioni:

- Si intende aggiornare i dati esistenti e sono necessari aggiornamenti al livello di colonna
- Alcuni dati non sono puliti
- Il database non è locale

**Nota:** Se si sta utilizzando la sincronizzazione dello spazio di ricerca di Product Advisor, è necessario utilizzare il metodo import SQL per il caricamento dei dati.

*Metodo delete:*

Il metodo delete viene utilizzato per eliminare i dati presenti nel documento XML di immissione dal database. Tale elemento deve contenere i valori per la chiave principale e l'indice univoco per la tabella. Se i dati da eliminare hanno dati in un'altra tabella dipendente da essa con l'opzione "cascade on delete" abilitata, vengono eliminati anche i dati dipendenti.

*Confronto dei metodi:*

- **Confronto dei metodi import SQL e load**

Il metodo import SQL verifica la consistenza dei dati, compresi i riferimenti esterni e consente di aggiornare i dati esistenti. Il metodo load non lo consente.

- **Confronto tra i metodi import ed import SQL**

I metodi import ed import SQL eseguono funzioni analoghe. Il metodo import è in genere più veloce, ma richiede spazio sul disco per i file temporanei.

Il metodo import, al contrario del metodo import SQL, può soltanto inserire o aggiornare le tabelle in cui è definita la chiave principale.

- **Confronto tra metodi basati sul prodotto database utilizzato**

I metodi import e load utilizzano programmi di utilità nativi ottimizzati per DB2, mentre il metodo import SQL utilizza le chiamate JDBC (che sono generiche per molti prodotti del database).

### **Considerazioni sulle prestazioni:**

Quando si utilizza Loader per caricare documenti di grandi dimensioni in un database, tener presente quanto segue:

- **Dimensioni della memoria riservata JVM (Java Virtual Machine)**

Per impostazione predefinita, la quantità massima di memoria assegnata a JVM è 64 MB. Se non viene aumentato tale valore, JVM potrebbe non avere sufficiente memoria a disposizione durante il processo di caricamento. La quantità massima di memoria assegnata a Java può essere modificata utilizzando l'opzione JVM -mx nel comando Java.

- **Registrazione traccia**

Il programma di registrazione della traccia può esaurire la memoria assegnata a JVM durante il caricamento di un documento XML di grandi dimensioni. Le informazioni della traccia vengono utilizzate principalmente per la correzione degli errori di un processo di esecuzione se questo non riesce correttamente. Se la traccia del processo di caricamento non è necessaria, è opportuno disattivarla. Quando la traccia viene disattivata, le prestazioni migliorano significativamente. La traccia viene disattivata modificando il documento XML di configurazione della registrazione. Per informazioni sulla modifica del documento XML di configurazione della registrazione, fare riferimento alla versione più recente del manuale *IBM WebSphere Commerce 5.4 Catalog Manager - Guida per l'utente*.

- **Conteggio del commit**

Il conteggio del commit predefinito per il Loader quando è operativo in modalità di importazione SQL è 1. Pertanto, per impostazione predefinita, le transazioni vengono eseguite per ogni aggiornamento o inserimento nel database. Per migliorare le prestazioni del pacchetto Loader per i documenti di grandi dimensioni, è necessario aumentare il conteggio del commit. Si consiglia un valore pari a 100; tuttavia è possibile aumentarlo a seconda della quantità di memoria fisica sul server, della dimensione del file di registrazione della transazione DBMS e così via.

E' possibile modificare il conteggio del commit per il pacchetto Loader utilizzando l'opzione *count* -commitcount per il comando Load (dove *count* è il numero delle istruzioni eseguite prima della transazione).

- **Configurazione della registrazione**

Un rallentamento nel caricamento dei dati può essere causato da una delle seguenti situazioni:

- L'utente che ha richiamato il pacchetto Loader non dispone dell'autorizzazione per la scrittura nella directory o per aggiornare il file specificato nel documento relativo alla configurazione della registrazione.
- La directory di ubicazione del file specificata nel documento relativo alla configurazione della registrazione non esiste.
- L'unità specificata come ubicazione del file nel documento di configurazione della registrazione non dispone di sufficiente spazio.

Una volta risolti questi problemi, è possibile cambiare la posizione specificata del file modificando il documento di configurazione della registrazione (per impostazione predefinita *WCALoggerConfig.xml*). Per informazioni sulla modifica del documento XML di configurazione della registrazione, fare riferimento alla versione più recente del manuale *IBM WebSphere Commerce 5.4 Catalog Manager - Guida per l'utente*.

Per informazioni dettagliate sull'impostazione e la personalizzazione dei file utilizzati per eseguire questo comando, fare riferimento alla versione più recente del manuale *IBM WebSphere Commerce 5.4 Catalog Manager - Guida per l'utente*.

## Comandi del pacchetto Loader per la trasformazione e l'estrazione dei dati

### Comando DTD Generate

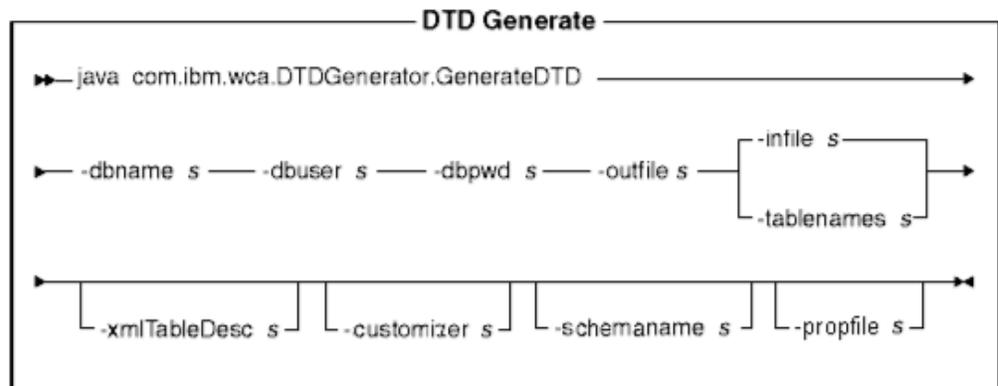
Questo comando crea una DTD da utilizzare con il pacchetto Loader. Questa DTD viene utilizzata durante tutto il processo di caricamento dei dati. A seconda del modo in cui si richiama il comando, DTD Generator può generare solo una DTD oppure una DTD e uno schema XML.

DTD Generator può creare una DTD basata sullo schema di database WebSphere Commerce. Se si utilizzano le DTD fornite con gli archivi negozio di esempio e non si modifica lo schema di database, non è necessario generare una DTD con DTD Generator. Le DTD fornite risiedono nella seguente directory:

- ▶ **NT** `unità:\WebSphere\CommerceServer\xml\sar`
- ▶ **2000** `unità:\Program Files\WebSphere\CommerceServer\xml\sar`
- ▶ **AIX** `/usr/WebSphere/CommerceServer/xml/sar`
- ▶ **Solaris** ▶ **Linux** `/opt/WebSphere/CommerceServer/xml/sar`
- ▶ **400** `/QIBM/ProdData/WebCommerce/xml/sar`

Si consiglia di utilizzare le DTD fornite. Se si personalizza uno schema di database, tuttavia, è necessario creare una nuova DTD o modificare quella fornita affinché soddisfi le modifiche apportate.

▶ **Windows** ▶ **AIX** ▶ **Solaris** ▶ **Linux**



#### Note:

1. Il grafico sopra riportato rappresenta principalmente un riferimento per i parametri del comando. Lo script o il file di comando fornito per questo comando e elencato in "Uso dei comandi e degli script del pacchetto Loader" a pagina 261 funge da wrapper per il comando Java effettivo e accetta gli stessi parametri; pertanto, si consiglia di utilizzare lo script o il file di comando invece di richiamare direttamente il comando Java.
2. I nomi file specificati come parametri per questo comando possono essere preceduti da percorsi relativi o assoluti.

## Valori dei parametri:

### **-dbname**

Il nome del database di destinazione

### **-dbuser**

Il nome dell'utente che si collega al database

### **-dbpwd**

La password dell'utente che si collega al database

### **-outfile**

Nome del file DTD di emissione (preferibilmente con l'estensione .dtd)

**-infile** Il nome di un file di immissione contenente un nome della tabella di database in ogni riga

### **-tablenames**

Nomi delle tabelle separati da virgole e racchiusi tra virgolette (" ")

### **-xmltabledesc**

Percorso del file dello schema XML da creare. Questo parametro è facoltativo.

### **-customizer**

Il nome del file properties di personalizzazione da utilizzare. Questo parametro è facoltativo. Il file properties di personalizzazione imposta il modo in cui funziona DTD Generator.

DB2ConnectionCustomizer.properties è il file predefinito. Il file delle proprietà del cliente può essere specificato come illustrato nel seguente esempio:

```
-customizer d:\WebSphere\CommerceServer\prop\dttdgen.properties
```

Se questo file si trova in una directory specificata nella variabile di ambiente del sistema classpath, lo stesso file può essere specificato come illustrato nel seguente esempio:

```
-customizer dttdgen
```

Per ulteriori informazioni sulla creazione e la specificazione di un nuovo file properties di personalizzazione, fare riferimento alla versione più recente del manuale *IBM WebSphere Commerce 5.4 Catalog Manager - Guida per l'utente*.

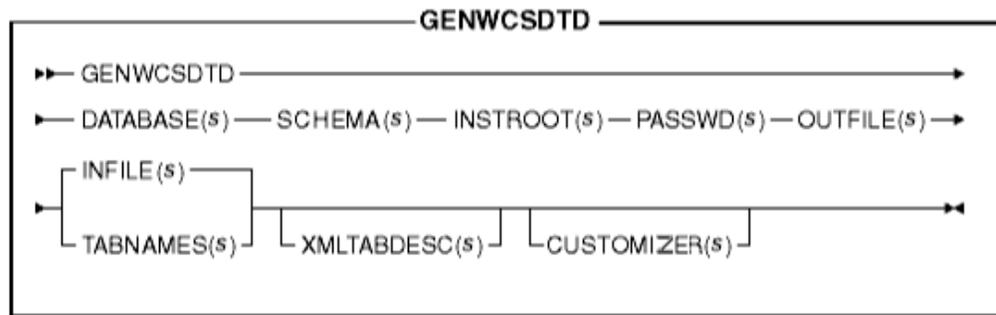
### **-schemaname**

Il nome dello schema del database di destinazione. Questo parametro è facoltativo.

Se questo parametro non è specificato durante l'esecuzione del comando, il comando cerca un'altra coppia nome=valore nel file properties di personalizzazione che specifica il valore di SchemaName. Se la coppia è presente nel file delle proprietà, il comando utilizza il valore specificato. Se per questo parametro non esiste specifica di file delle proprietà o di riga comandi, il comando utilizza per impostazione predefinita il nome dell'utente del database.

### **-profile**

I file properties in cui il testo di aiuto, i valori predefiniti e le informazioni di descrizione dei campi possono essere memorizzati per una descrizione del modulo di Web editor. Questo parametro è facoltativo.



**Nota:** I nomi file specificati come parametri per questo comando possono essere preceduti da percorsi relativi o assoluti.

#### Valori dei parametri:

##### DATABASE

Il nome del database di destinazione, come visualizzato nella directory del database relazionale

##### SCHEMA

Il nome dello schema del database di destinazione, uguale al nome dell'istanza

##### INSTROOT

Il nome completo del percorso root dell'istanza di WebSphere Commerce, ad esempio /QIBM/UserData/WebCommerce/instances/nome\_istanza

##### PASSWD

Password dell'istanza di WebSphere Commerce

##### OUTFILE

Nome del file DTD di emissione (preferibilmente con l'estensione .dtd)

##### INFILE

Il nome di un file di immissione contenente un nome della tabella di database in ogni riga

##### TABNAMES

Nomi delle tabelle separati da virgole e racchiusi tra virgolette (" ")

##### XMLTABDESC

Percorso del file dello schema XML da creare. Questo parametro è facoltativo.

##### CUSTOMIZER

Il nome del file properties di personalizzazione da utilizzare. Questo parametro è facoltativo. Il file properties di personalizzazione imposta il modo in cui funziona DTD Generator. Il file predefinito è ISeries\_GENWCSDTD\_Customizer.properties. Il file properties del cliente può essere specificato come illustrato nel seguente esempio:

```
CUSTOMIZER(/wc/prop/dtdgen.properties)
```

Se questo file si trova in una directory specificata nella variabile di ambiente del sistema classpath, lo stesso file può essere specificato come illustrato nel seguente esempio:

```
CUSTOMIZER(dtdgen)
```

Per ulteriori informazioni sulla creazione e la specificazione di un nuovo file properties di personalizzazione, fare riferimento alla versione più recente del manuale *IBM WebSphere Commerce 5.4 Catalog Manager - Guida per l'utente*.

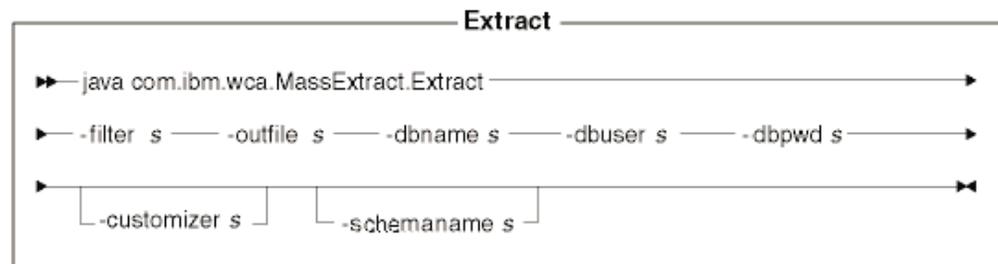
Per informazioni dettagliate sull'impostazione e la personalizzazione dei file utilizzati per eseguire questo comando, fare riferimento alla versione più recente del manuale *IBM WebSphere Commerce 5.4 Catalog Manager - Guida per l'utente*.

## Comando Extract

Questo comando consente di estrarre da un database un sottoinsieme di dati selezionato in formato file XML.

Per estrarre dei dati da un database mediante Extractor, è necessario specificare i dati che si desidera estrarre utilizzando un file filtro di estrazione. Il filtro di estrazione che si utilizza dipende dal tipo di dati che si desidera estrarre.

Windows AIX Solaris Linux



### Note:

1. Il grafico sopra riportato rappresenta principalmente un riferimento per i parametri del comando. Lo script o il file di comando fornito per questo comando e elencato in "Uso dei comandi e degli script del pacchetto Loader" a pagina 261 funge da wrapper per il comando Java effettivo e accetta gli stessi parametri; pertanto, si consiglia di utilizzare lo script o il file di comando invece di richiamare direttamente il comando Java.
2. I nomi file specificati come parametri per questo comando possono essere preceduti da percorsi relativi o assoluti.

### Valori dei parametri:

**-filter** Il nome del file del filtro di estrazione

#### **-outfile**

Il nome del file XML di emissione in cui vengono memorizzati i dati estratti

#### **-dbname**

Il nome del database dal quale vengono estratti i dati

#### **-dbuser**

Il nome utente del database dal quale vengono estratti i dati

#### **-dbpwd**

La password associata al nome utente del database dal quale vengono estratti i dati

#### **-customizer**

Il nome del file properties di personalizzazione da utilizzare. Il file

properties di personalizzazione imposta il modo in cui funziona Extractor. DB2ConnectionCustomizer.properties è il file predefinito. Il file properties del cliente può essere specificato come illustrato nel seguente esempio:

```
-customizer d:\WebSphere\CommerceServer\prop\extract.properties
```

Se questo file si trova in una directory specificata nella variabile di ambiente del sistema classpath, lo stesso file può essere specificato come illustrato nel seguente esempio:

```
-customizer extract
```

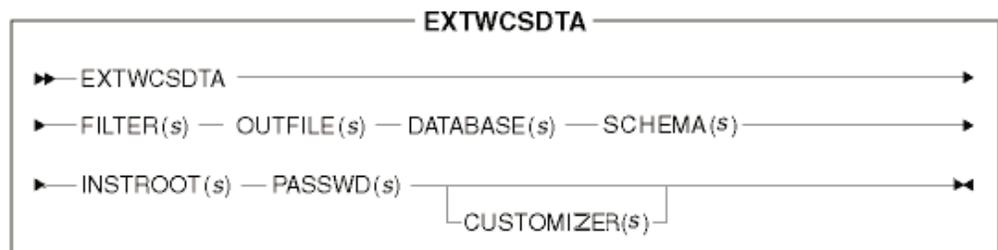
Per ulteriori informazioni sulla creazione e la specificazione di un nuovo file properties di personalizzazione, fare riferimento alla versione più recente del manuale *IBM WebSphere Commerce 5.4 Catalog Manager - Guida per l'utente*.

#### **-schemaname**

Il nome dello schema di database dal quale vengono estratti i dati. Questo parametro è facoltativo.

Se questo parametro non è specificato durante l'esecuzione del comando, il comando cerca un'altra coppia nome=valore nel file properties di personalizzazione che specifica il valore di SchemaName. Se la coppia è presente nel file delle proprietà, il comando utilizza il valore specificato. Se per questo parametro non esiste specifica di file delle proprietà o di riga comandi, il comando utilizza per impostazione predefinita il nome schema della tabella nel database.

▶ 400



**Nota:** I nomi file specificati come parametri per questo comando possono essere preceduti da percorsi relativi o assoluti.

#### **Valori dei parametri:**

##### **FILTER**

Il nome del file del filtro di estrazione

##### **OUTFILE**

Il nome del file XML di emissione in cui vengono memorizzati i dati estratti

##### **DATABASE**

Il nome del database dal quale vengono estratti i dati come visualizzato nella directory del database relazionale

##### **SCHEMA**

Il nome dello schema del database dal quale vengono estratti i dati; questo nome è uguale al nome istanza

## INSTROOT

Il nome completo del percorso root dell'istanza di WebSphere Commerce, ad esempio /QIBM/UserData/WebCommerce/instances/*nome\_istanza*

## PASSWD

Password dell'istanza di WebSphere Commerce

## CUSTOMIZER

Il nome del file properties di personalizzazione da utilizzare. Il file properties di personalizzazione imposta il modo in cui funziona Extractor. Il file predefinito è ISeries\_EXTWCSDTA\_Customizer.properties. Il file properties del cliente può essere specificato come illustrato nel seguente esempio:

```
CUSTOMIZER(/wc/prop/extract.properties)
```

Se questo file si trova in una directory specificata nella variabile di ambiente del sistema classpath, lo stesso file può essere specificato come illustrato nel seguente esempio:

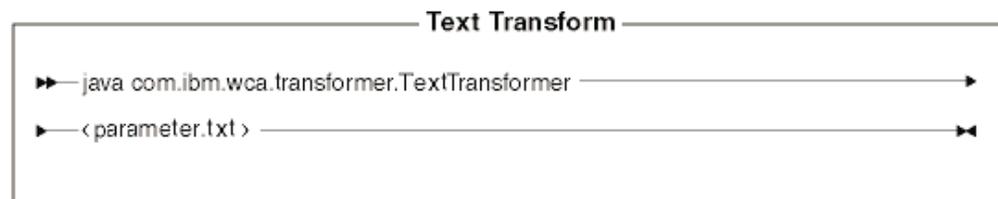
```
CUSTOMIZER(extract)
```

Per ulteriori informazioni sulla creazione e la specificazione di un nuovo file properties di personalizzazione, fare riferimento alla versione più recente del manuale *IBM WebSphere Commerce 5.4 Catalog Manager - Guida per l'utente*.

Per ulteriori informazioni su questo comando, fare riferimento alla versione più recente del manuale *IBM WebSphere Commerce 5.4 Catalog Manager - Guida per l'utente*.

## Comando Text Transform

Questo comando converte i dati da formato variabile delimitato da caratteri in formato XML.



**Nota:** Il grafico sopra riportato rappresenta principalmente un riferimento per i parametri del comando. Lo script o il file di comando fornito per questo comando ed elencato in "Uso dei comandi e degli script del pacchetto Loader" a pagina 261 funge da wrapper per l'effettivo comando Java e accetta gli stessi parametri; pertanto, si consiglia di utilizzare il file di comando o lo script invece di richiamare il comando Java direttamente.

### Valori dei parametri:

I valori di seguito riportati vengono specificati e separati da virgole in un file di parametri (*parametro.txt*):

#### input file

Il nome del file da convertire

**schema file**

Il nome del file di schema XML da utilizzare nella conversione

**output file**

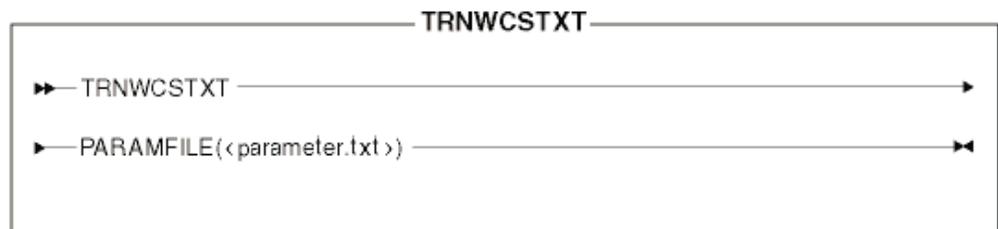
Il nome del file di emissione nel quale verranno memorizzati i dati convertiti

**transformation method**

Il metodo da utilizzare nell'aggiunta dei dati al file di emissione. Specificare **Crea** se deve essere creato un nuovo file oppure specificare **Aggiungi** se i dati di emissione devono essere aggiunti ad un file di dati esistente.

Questo file è detto anche file di "manifesto" o "comando". Può contenere più righe di quattro parametri ognuna.

▶ 400

**Valori dei parametri:**

I valori di seguito riportati vengono specificati e separati da virgole in un file di parametri (*parametro.txt*):

**input file**

Il nome del file da convertire

**schema file**

Il nome del file di schema XML da utilizzare nella conversione

**output file**

Il nome del file di emissione nel quale verranno memorizzati i dati convertiti

**transformation method**

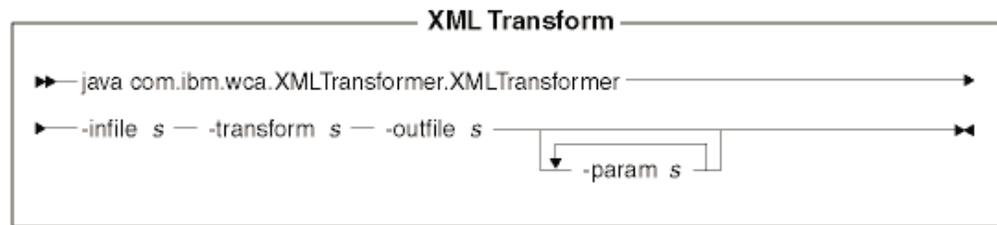
Il metodo da utilizzare nell'aggiunta dei dati al file di emissione. Specificare **Crea** se deve essere creato un nuovo file oppure specificare **Aggiungi** se i dati di emissione devono essere aggiunti ad un file di dati esistente.

**Nota:** Questo file è detto anche file di "manifesto" o "comando".

Per ulteriori informazioni su questo comando, fare riferimento alla versione più recente del manuale *IBM WebSphere Commerce 5.4 Catalog Manager - Guida per l'utente*.

**Comando XML Transform**

Questo comando converte un file XML in un formato diverso.



**Note:**

1. Il grafico sopra riportato rappresenta principalmente un riferimento per i parametri del comando. Lo script o il file di comando fornito per questo comando e elencato in "Uso dei comandi e degli script del pacchetto Loader" a pagina 261 funge da wrapper per il comando Java effettivo e accetta gli stessi parametri; pertanto, si consiglia di utilizzare lo script o il file di comando invece di richiamare direttamente il comando Java.
2. I nomi file specificati come parametri per questo comando possono essere preceduti da percorsi relativi o assoluti.

**Valori dei parametri:**

**-infile** Il nome del file da convertire

**-transform**

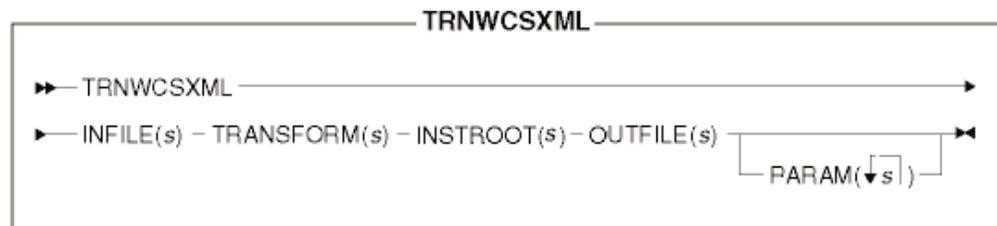
Il nome del file della regola di associazione XSL di conversione

**-outfile**

Il nome del file XML di emissione nel quale verranno memorizzati i dati convertiti

**-param**

Il parametro da trasferire al file di regole di associazione XSL. *Questo parametro è facoltativo.* Questo parametro può essere specificato più volte per trasmettere più coppie nome=valore.



**Nota:** I nomi file specificati come parametri per questo comando possono essere preceduti da percorsi relativi o assoluti.

**Valori dei parametri:**

**INFILE**

Il nome del file da convertire

**TRANSFORM**

Il nome del file della regola di associazione XSL di conversione

**INSTROOT**

Il nome completo del percorso root dell'istanza di WebSphere Commerce, ad esempio /QIBM/UserData/WebCommerce/instances/*nome\_istanza*

**OUTFILE**

Il nome del file XML di emissione nel quale verranno memorizzati i dati convertiti

**PARAM**

Il parametro da trasferire al file di regole di associazione XSL. *Questo parametro è facoltativo*. La stringa può contenere più valori per trasferire più coppie nome=valore.

Per ulteriori informazioni su questo comando, fare riferimento alla versione più recente del manuale *IBM WebSphere Commerce 5.4 Catalog Manager - Guida per l'utente*.

## Strumenti relativi ai comandi del pacchetto Loader

### Strumento Text Transformation

Text Transformation consente di eseguire una conversione di dati da un formato variabile delimitato da caratteri a un formato XML mediante il comando Text Transform. Vengono fornite le seguenti viste:

1. La visualizzazione per la modifica dello schema del testo consente di creare e modificare il file di schema XML da utilizzare in una conversione.
2. La Vista per la modifica comando trasformazione consente di creare e modificare i comandi effettivi utilizzati per eseguire il processo di trasformazione.
3. La Vista processo di trasformazione consente di avviare il processo di trasformazione.

Per ulteriori informazioni su questo strumento, fare riferimento alla versione più recente del manuale *IBM WebSphere Commerce 5.4 Catalog Manager - Guida per l'utente*.

### XSL editor

Lo strumento XML Transformer utilizza il linguaggio XSL per definire le regole per la trasformazione di un file XML in un altro file XML. La funzione di associazione di XSL editor fornisce un'interfaccia visiva con cui è possibile stabilire l'associazione tra un elemento in una DTD di origine e un elemento in una DTD di destinazione. Date due DTD, è possibile sviluppare regole XSL per stabilire il modo in cui un file XML conforme alla prima DTD (origine) venga trasformato in un file conforme alla seconda DTD (destinazione).

Per ulteriori informazioni su questo strumento, fare riferimento alla versione più recente del manuale *IBM WebSphere Commerce 5.4 Catalog Manager - Guida per l'utente*.

### Web editor

Web editor consente di creare, eliminare e modificare i dati di catalogo con un browser Web. I formati dei dati per la visualizzazione e l'aggiornamento delle informazioni sono fondamentali in Web editor. Nel caso più semplice, i formati corrispondono alle tabelle del database di WebSphere Commerce Server. Lo sviluppatore del negozio può scegliere di utilizzare i formati predefiniti o di personalizzare i formati disponibili.

Per ulteriori informazioni su questo strumento, fare riferimento alla versione più recente del manuale *IBM WebSphere Commerce 5.4 Catalog Manager - Guida per l'utente*.

---

## Caricamento dei dati del negozio

Questa sezione fornisce esempi su come caricare i dati nel database WebSphere mediante le utilità della riga comandi del pacchetto Loader.

**Note:**

1. Gli esempi riportati in questa sezione sono eseguiti in un ambiente Windows NT. Per informazioni sull'esecuzione di questi comandi in altri ambienti, fare riferimento alla versione più recente del manuale *IBM WebSphere Commerce 5.4 Catalog Manager - Guida per l'utente*.
2. Sebbene le utilità di riga comandi del pacchetto Loader supportino DB2, DB2 per iSeries e Oracle, solo i comandi e le opzioni per DB2 sono incluse nei seguenti esempi. Se si utilizza un database diverso da DB2, assicurarsi di modificare i file properties di personalizzazione come indicato nella versione più recente del manuale *IBM WebSphere Commerce 5.4 Catalog Manager - Guida per l'utente*.

Per informazioni sul processo di caricamento dei gruppi di risorse database di WebSphere Commerce, fare riferimento a Capitolo 28, "Caricamento dei gruppi di risorse di database di WebSphere Commerce" a pagina 271 e "Caricamento dei gruppi di risorse database" a pagina 279.

## Uso dei comandi e degli script del pacchetto Loader

Per eseguire i comandi del pacchetto Loader, utilizzare gli script o i comandi nella seguente directory di WebSphere Commerce:

- ▶ **NT** *unità:*\WebSphere\CommerceServer\bin
- ▶ **2000** *unità:*\Program Files\WebSphere\CommerceServer\bin
- ▶ **AIX** /usr/WebSphere/CommerceServer/bin
- ▶ **Solaris** ▶ **Linux** /opt/WebSphere/CommerceServer/bin
- ▶ **400** QWEEBCOMM native library

Gli script e i comandi sono riportati di seguito:

- ▶ **Windows**
  - dtdgen.cmd** Comando DTD Generate
  - idresgen.cmd** Comando ID Resolve
  - massload.cmd** Comando Load
  - massextract.cmd** Comando Extract
  - txtransform.cmd** Comando Text Transform
  - xmltransform.cmd** Comando XML Transform
- ▶ **AIX** ▶ **Solaris** ▶ **Linux**
  - dtdgen.sh** Script shell DTD Generate
  - idresgen.sh** Script shell ID Resolve
  - massload.sh** Script shell Load
  - massextract.sh** Script shell Extract
  - txtransform.sh** Script shell Text Transform
  - xmltransform.sh** Script shell XML Transform
- ▶ **400**
  - GENWCSDTD** Comando DTD Generate
  - RESWCSID** Comando ID Resolve
  - LODWCSDTA** Comando Load
  - EXTWCSDTA** Comando Extract
  - TRNWCSTXT** Comando Text Transform
  - TRNWCSXML** Comando XML Transform

## Esempi di risoluzione degli identificativi

Gli esempi di risoluzione degli identificativi descritti in questa sezione utilizzano i file di risorse negozio del negozio di esempio  ToolTechh.

Poiché questo esempio si basa sul caricamento di nuovi dati nel database di WebSphere Commerce Server, viene utilizzato il metodo load.

Se successivamente è necessario modificare alcuni elementi del documento XML, è possibile farlo con il metodo update. Questo metodo è più veloce del metodo load perché non vengono allocati nuovi identificativi. Con il metodo update, per individuare l'identificativo viene eseguita una query di database. Se l'identificativo non viene rilevato, viene restituito un messaggio di errore. Fare riferimento alla discussione che comincia a pagina 240 per ulteriori informazioni sul funzionamento di questo processo.

Se il file XML di immissione contiene sia elementi che esistono sia elementi che non esistono nel database, utilizzare il metodo misto. Con questo metodo, viene innanzitutto eseguita una ricerca nel database, quindi, se il record non viene rilevato, viene assegnato un identificativo all'elemento. In caso di dubbio, utilizzare il metodo misto. Sebbene i metodi load e update siano più veloci, il file XML risultò creato con il metodo misto ha maggiori probabilità di caricamento senza errori.

Per una discussione del funzionamento di ID Resolver, fare riferimento a “Metodi per il comando ID Resolve” a pagina 241.

### Risoluzione degli identificativi in file XML con alias interni

Per risolvere gli identificativi mediante alias interni prima di caricare i dati nel database WebSphere Commerce Server, eseguire il comando ID Resolve come indicato nel seguente esempio:

1. Assicurarsi che il percorso includa la directory contenente il comando o lo script ID Resolve appropriato, elencati in “Uso dei comandi e degli script del pacchetto Loader” a pagina 261.

In questo esempio, viene utilizzato il comando ID Resolve `idresgen.cmd`.

2. Creare una directory di lavoro.

In questo esempio, creare la seguente directory:  
`c:\WebSphere\CommerceServer\runtime\test`.

3. Assicurarsi che il file XML di immissione e i file DTD di riferimento abbiano un percorso riconoscibile da ID Resolver.

Ad esempio, effettuare le seguenti operazioni:

- a. Copiare `c:\WebSphere\CommerceServer\samplestores\ToolTech\ToolTech_en_US_fr_FR.sar` in `c:\WebSphere\CommerceServer\runtime\test`.
- b. Dal prompt dei comandi di Windows, immettere il seguente comando:  
`cd c:\WebSphere\CommerceServer\runtime\test`
- c. Dal prompt dei comandi di Windows, immettere il seguente comando:  
`jar -xvf ToolTech_en_US_fr_FR.sar`

I file XML del negozio di esempio ToolTech  vengono estratti in `c:\WebSphere\CommerceServer\runtime\test\data`.

- d. Dal prompt dei comandi di Windows, immettere il seguente comando:

```
cp c:\WebSphere\CommerceServer\xml\sar\store.dtd
c:\WebSphere\CommerceServer\runtime\test\data
```

Questo comando copia il file store.dtd in  
c:\WebSphere\CommerceServer\runtime\test\data.

- e. Dal prompt dei comandi di Windows, immettere il seguente comando:

```
cp c:\WebSphere\CommerceServer\xml\sar\DBLoadMacros.dtd
c:\WebSphere\CommerceServer\runtime\test\data
```

Questo comando copia il file DBLoadMacros.dtd in  
c:\WebSphere\CommerceServer\runtime\test\data.

- f. Dal prompt dei comandi di Windows, immettere il seguente comando:

```
cp c:\WebSphere\CommerceServer\xml\sar\fulfillment.dtd
c:\WebSphere\CommerceServer\runtime\test\data
```

Questo comando copia il file fulfillment.dtd in  
c:\WebSphere\CommerceServer\runtime\test\data.

4. Assicurarsi che lo schema di WebSphere sia caricato nel database insieme ai dati di bootstrap necessari creando un'istanza di database WebSphere Commerce Server appropriata.

**Nota:** Per informazioni sulla creazione di un'istanza, fare riferimento alla guida l'installazione di WebSphere per il sistema operativo utilizzato.

L'istanza di database WebSphere Commerce Server utilizzata da questo esempio è denominata *mall*. Le chiavi primaria ed esterna sono ricavate dalle tabelle KEYS e SUBKEYS di questo database; pertanto, se il database non viene caricato correttamente ID Resolver non può risolvere gli identificativi.

5. Nel file store.xml è presente il seguente elemento:

```
<storeent
 STOREENT_ID="@storeent_id_1"
 MEMBER_ID="&MEMBER_ID;"
 TYPE="S"
 IDENTIFIER="ToolTech"
 SETCURR="USD"
/>
```

Questo elemento di store.xml è associato alla tabella storeent del database e gli attributi STOREENT\_ID MEMBER\_ID, TYPE, IDENTIFIER e SETCURR alle colonne di tale tabella. La specificazione @storeent\_id\_1 è un alias interno del valore dell'attributo STOREENT\_ID e &MEMBER\_ID; è un parametro entità. Il valore dell'entità &MEMBER\_ID; deve essere sostituito prima di poter essere caricato con Loader. Il valore di &MEMBER\_ID; viene definito nel file di macro DBLoadMacros.dtd e viene sostituito da quel file. Quando ID Resolver rileva @storeent\_id\_1, controlla nella tabella di chiavi primarie se storeent è presente. Poiché è una tabella primaria, storeent è presente. ID Resolver richiama il contatore per quella tabella, lo incrementa e lo sostituisce all'alias interno. Altre voci dello stesso tipo presenti nel file store.xml vengono elaborate nello stesso modo.

6. Dal prompt dei comandi di Windows, immettere il seguente comando:
- ```
idresgen -dbname mall -dbuser db2admin -dbpwd db2admin -infile store.xml
-outfile c:\WebSphere\CommerceServer\runtime\test\data\store1.xml -method load
```

Il primo frammento XML di emissione in store1.xml è simile al seguente:

```
<storeent
  STOREENT_ID="10001"
  MEMBER_ID="-2001"
  TYPE="S"
  IDENTIFIER="ToolTech"
  SETCCURR="USD"
/>
```

Il secondo frammento XML in store1.xml è simile al seguente:

```
<store
  STORE_ID="10001"
  DIRECTORY="ToolTech"
  FFMCENTER_ID=""
  LANGUAGE_ID="-1"
  STOREGRP_ID="-1"
  ALLOCATIONGOODFOR="43200"
  BOPMPADFACTOR="0"
  DEFAULTBOFFSET="2592000"
  FFMCELECTIONFLAGS="0"
  MAXBOFFSET="7776000"
  REJECTEDORDEXPIRY="259200"
  RTNFFMCTR_ID=""
  PRICEREFFLAGS="0"
  STORETYPE="B2B"
/>
```

Gli attributi FFMCENTER_ID e RTNFFMCTR_ID non sono stati risolti. Invece di essere risolto, l'alias interno @ffmcenter_id_1 è stato sostituito con virgolette vuote (" "). E' un errore. Non sarà possibile caricare i dati correttamente mediante Loader a causa di una violazione di una limitazione relativa alla chiave esterna. ID Resolver non è stato in grado di risolvere questo alias interno perché la tabella FFMCENTER non è stata elaborata prima di utilizzare un alias che vi faceva riferimento. Per risolvere il problema, scegliere una delle seguente opzioni:

- Opzione 1:
 - a. Immettere il comando di seguito riportato per eseguire ID Resolver sul file fulfillment.xml (dove è definita la tabella FFMCENTER):

```
idresgen -dbname mall -dbuser db2admin -dbpwd db2admin
-infile fulfillment.xml -outfile
c:\WebSphere\CommerceServer\runtime\test\data\fulfillment1.xml
-method load
```

L'elemento risolto nel file di emissione fulfillment1.xml è simile al seguente:

```
<fulfillment-asset>
<ffmcenter
  FFMCENTER_ID="10001"
  MEMBER_ID="-2001"
  NAME="ToolTech Home"
  DEFAULTSHIPOFFSET="0"
  MARKFORDELETE="0"
/>
</fulfillment-asset>
```

- b. Ricavare la chiave FFMCENTER_ID dal file di emissione risultante (fulfillment1.xml) e sostituire tale chiave a tutte le ricorrenze di

@ffmcenter_id_1 della copia di lavoro di store.xml in
c:\WebSphere\CommerceServer\runtime\test\data.

c. Immettere il seguente comando:

```
idresgen -dbname mall -dbuser db2admin -dbpwd db2admin -infile store.xml  
-outfile c:\WebSphere\CommerceServer\runtime\test\data\store1.xml  
-method load
```

Gli elementi completamente risolti del file di emissione store1.xml sono simili ai seguenti:

```
<store-asset>  
<storeent  
  STOREENT_ID="10151"  
  MEMBER_ID="-2001"  
  TYPE="S"  
  IDENTIFIER="ToolTech"  
  SETCCURR="USD"  
</storeent>  
<store  
  STORE_ID="10151"  
  DIRECTORY="ToolTech"  
  FFMCENTER_ID="10001"  
  LANGUAGE_ID="-1"  
  STOREGRP_ID="-1"  
  ALLOCATIONGOODFOR="43200"  
  BOPMPADFACTOR="0"  
  DEFAULTBOFFSET="2592000"  
  FFMCELECTIONFLAGS="0"  
  MAXBOFFSET="7776000"  
  REJECTEDORDEXPIRY="259200"  
  RTNFFMCTR_ID="10001"  
  PRICEREFFLAGS="0"  
  STORETYPE="B2B"  
</store>  
<vendor  
  VENDOR_ID="10001"  
  STOREENT_ID="10151"  
  VENDORNAME="Tooltech Vendor"  
  MARKFORDELETE="0"  
</vendor>  
<dispentrel  
  AUCTIONSTATE="0"  
  CATENTRY_ID="0"  
  CATENTTYPE_ID="ProductBean"  
  DEVICEFMT_ID="-1"  
  DISPENTREL_ID="10001"  
  MBRGRP_ID="0"  
  PAGENAME="CatalogProductDisplay.jsp"  
  STOREENT_ID="10151"  
  RANK="0"  
</dispentrel>  
<dispentrel  
  AUCTIONSTATE="0"  
  CATENTRY_ID="0"  
  CATENTTYPE_ID="ItemBean"  
  DEVICEFMT_ID="-1"  
  DISPENTREL_ID="10002"  
  MBRGRP_ID="0"  
  PAGENAME="CatalogItemDisplay.jsp"  
  STOREENT_ID="10151"  
  RANK="0"  
</dispentrel>  
<dispcgprel  
  CATGROUP_ID="0"  
  DEVICEFMT_ID="-1"  
  DISPCGPREL_ID="10001"
```

```

        MBRGRP_ID="0"
        PAGENAME="CatalogCategories.jsp"
        STOREENT_ID="10151"
        RANK="0"
    />
    <invadjcode
        ADJUSTCODE="PCNT"
        INVADJCODE_ID="10001"
        MARKFORDELETE="0"
        STOREENT_ID="10151"
    />
    <invadjcode
        ADJUSTCODE="SPLG"
        INVADJCODE_ID="10002"
        MARKFORDELETE="0"
        STOREENT_ID="10151"
    />
    <invadjcode
        ADJUSTCODE="DISC"
        INVADJCODE_ID="10003"
        MARKFORDELETE="0"
        STOREENT_ID="10151"
    />
    <rtreason
        REASONTYPE="C"
        RTNREASON_ID="10001"
        STOREENT_ID="10151"
        MARKFORDELETE="0"
        CODE="WPR"
    />
    <rtreason
        REASONTYPE="B"
        RTNREASON_ID="10002"
        STOREENT_ID="10151"
        MARKFORDELETE="0"
        CODE="DEF"
    />
    <rtreason
        REASONTYPE="M"
        RTNREASON_ID="10003"
        STOREENT_ID="10151"
        MARKFORDELETE="0"
        CODE="ERR"
    />
    <rtreason
        REASONTYPE="M"
        RTNREASON_ID="10004"
        STOREENT_ID="10151"
        MARKFORDELETE="0"
        CODE="WPS"
    />
</store-asset>

```

- Opzione 2:

- a. Unire i file `fulfillment.xml` e `store.xml` aggiungendo il contenuto univoco in `fulfillment.xml` (incluso il riferimento a `fulfillment.dtd`) a `store.xml`, assicurandosi che l'elemento `ffmcenter` di seguito mostrato preceda l'elemento `store`.

```

<ffmcenter
    FFMCENTER_ID="@ffmcenter_id_1"
    MEMBER_ID="&MEMBER_ID;"
    NAME="ToolTech Home"
    DEFAULTBOOFFSET="0"
    MARKFORDELETE="0"
/>

```

- b. Eseguire ID Resolver sul file risultante dall'unione.

- Opzione 3: Caricare il gruppo di risorse negozio mediante il processo descritto in “Caricamento dei gruppi di risorse database” a pagina 279.

Specifica di un file properties con ID Resolver

E' possibile modificare il modo in cui ID Resolver risolve gli identificativi utilizzando il parametro `-propfile`. Il file properties predefinito è `IdResolveKeys.properties`, ma è possibile modificarlo o specificarne uno nuovo quando si richiama il comando di ID Resolve.

-     `IdResolveKeys.properties` si trova nella seguente directory:

-  `unità:\WebSphere\CommerceServer\properties`
-  `unità:\Program Files\WebSphere\CommerceServer\properties`
-  `/usr/WebSphere/CommerceServer/properties`
-   `/opt/WebSphere/CommerceServer/properties`

Se quando si esegue ID Resolver, questo file non viene inserito nella directory corrente, è possibile inserirlo in una directory definita nella variabile di ambiente del classpath. E' anche possibile specificare il percorso completo per questo file.

-  Per modificare il file `IdResolveKeys.properties`, copiarlo dalla directory `/QIBM/ProdData/WebCommerce/properties`, salvarlo nella directory `/rootist/xml`, quindi apportare tutte le modifiche necessarie al nuovo file.

Nota: La directory precedente si trova nel classpath utilizzato dal comando `RESWCSID`.

La specificazione del file properties ha la precedenza sull'uso degli alias interni.

Viene di seguito riportato un esempio di frammento XML del file `store.xml`:

```
<store
  STORE_ID="@storeent_id_1"
  DIRECTORY="ToolTech"
  FFMCENTER_ID="@ffmcenter_id_1"
  LANGUAGE_ID="&en_US;"
  STOREGRP_ID="-1"
  ALLOCATIONGOODFOR="43200"
  BOPMPADFACTOR="0"
  DEFAULTB00FFSET="2592000"
  FFMCELECTIONFLAGS="0"
  MAXB00FFSET="7776000"
  REJECTEDORDEXPIRY="259200"
  RTNFFMCTR_ID="@ffmcenter_id_1"
  PRICEREFFLAGS="0"
  STORETYPE="B2B"
/>
```

Se ID Resolver è stato eseguito con il parametro `-propfile` specificato come `c:\WebSphere\CommerceServer\runtime\test\data\myPropFile` e il file specificato, `myPropFile.properties`, contiene le seguenti voci:

```
NAMEDELIMETER=@
SELECTDELIMETER=:
FFMCENTER=@FFMCENTER_ID@MEMBER_ID:10051 -2001
```

ID Resolver ricerca nel database la tabella `FFMCENTER` con una clausola `where` di `10051` e `-2001` quando viene elaborato l'elemento `store`. L'indice restituito per questo valore viene poi utilizzato per risolvere l'identificativo per `FFMCENTER_ID`.

Per ulteriori informazioni sull'uso di questo comando, fare riferimento a "Comando ID Resolve" a pagina 234.

Esempio di caricamento dei dati

Una volta risolti gli identificativi nel file XML, è possibile caricare i dati nel database WebSphere Commerce Server.

Nota: Se gli identificativi sono stati risolti correttamente nei dati XML, il file XML di origine non deve contenere i seguenti elementi:

- parole precedute dal simbolo at (@)
- parole precedute dal simbolo di E commerciale (&)
- identificativi con virgolette vuote (" ")

La presenza di uno di questi elementi indica che il file XML non può ancora essere caricato.

L'esempio di caricamento dati descritto in questa sezione utilizza il file `fulfillment1.xml` risolto in "Esempi di risoluzione degli identificativi" a pagina 262.

Per caricare i dati nel database WebSphere Commerce Server, eseguire il comando Load come mostrato nell'esempio:

1. Assicurarsi che il percorso includa la directory contenente il comando o lo script ID Resolve appropriato, elencati in "Uso dei comandi e degli script del pacchetto Loader" a pagina 261.

In questo esempio, utilizzare `massload.cmd`.

2. Creare una directory di lavoro.

In questo esempio, utilizzare la directory

`c:\WebSphere\CommerceServer\runtime\test\data` creata in "Esempi di risoluzione degli identificativi" a pagina 262.

3. Assicurarsi che il file XML di immissione risieda in un percorso riconoscibile dal programma Loader.

Per questo esempio, assicurarsi che il file di emissione `fulfillment1.xml` creato in "Esempi di risoluzione degli identificativi" a pagina 262 risieda in `c:\WebSphere\CommerceServer\runtime\test\data`.

4. Passare alla directory di lavoro.

In questo esempio, immettere il seguente comando da un prompt dei comandi Windows:

```
cd c:\WebSphere\CommerceServer\runtime\test\data
```

5. Non dimenticare di eseguire una copia di backup del database WebSphere Commerce Server per poterlo ripristinare in caso di errori irreversibili.

6. Eseguire il comando Load sul file XML risolto per caricare i dati nel database di destinazione.

In questo esempio, immettere il seguente comando da un prompt dei comandi Windows:

```
massload -dbname mall -dbuser db2admin -dbpwd db2admin -infile  
c:\WebSphere\CommerceServer\runtime\test\data\fulfillment1.xml  
-method sqlimport -commitcount 50
```

Anche se gli elementi da caricare sono meno di 50, in questo esempio il valore del parametro `-commitcount` è 50 per motivi di prestazioni. Per impostazione predefinita, il conteggio del commit è pari a 1. Se si utilizza questo valore, viene eseguita un'operazione di commit per ogni record scritto nel database.

L'impostazione del numero su 50 nell'esempio sopra riportato assicura che l'I/E nel database si verifichi una sola volta se il caricamento è corretto e che nulla venga scritto nel database se si verificano errori. Se i dati da caricare sono

numerosi, tuttavia, si raccomanda di non assegnare al conteggio di commit un valore pari al numero di elementi per i seguenti motivi:

- Un valore di commit alto causa un uso elevato della memoria.
- Quando il numero di assegnazione è più piccolo del numero di elementi, almeno alcuni dati vengono scritti nel database. A seconda del valore di `-maxerror`, un valore più piccolo di `-commitcount` assicura che alcuni dati vengano scritti nel database prima che venga superato il numero massimo errori e terminato lo strumento. Il valore predefinito per `-maxerror` è 1.

L'impostazione predefinita dell'opzione `-noprimary` è `error` e fa in modo che lo strumento segnali gli errori e termini quando le chiavi primarie mancano.

Poiché in questi esempi le risorse negozio non vengono caricate per l'uso con Servizi del negozio come descritto in "Sequenza di caricamento delle risorse database" a pagina 271, il file `store2.xml` creato in "Esempi di risoluzione degli identificativi" a pagina 262 potrebbe violare le limitazioni di integrità di alcune tabelle. Se si tenta di caricare `store2.xml` senza modifiche mediante il metodo `load` e vengono violate le limitazioni, il database entra in uno stato di sospensione. Per semplicità, quindi, questo esempio di utilizzo del comando `Load` si basa sulla versione risolta del file `fulfillment.xml` la cui sola chiave esterna è quella di `MEMBER_ID` definita nel negozio di esempio. In questo esempio il file `fulfillment1.xml` risolto viene caricato come emissione in "Esempi di risoluzione degli identificativi" a pagina 262 e viene utilizzato il metodo `SQL import`. Se non si è certi che i contenuti del file XML sono puliti, utilizzare il metodo `SQL import` come mostrato in questo esempio con i parametri `-commitcount` e `-maxerror` impostati correttamente in modo che qualsiasi violazione delle limitazioni del database sia indicata senza modificare il database e danneggiarne l'integrità.

Per ulteriori informazioni sull'uso di questo comando, fare riferimento a "Comando `Load`" a pagina 242.

Capitolo 28. Caricamento dei gruppi di risorse di database di WebSphere Commerce

Se non si desidera creare tutte le risorse database e inserirle nel file di archivio negozio prima della pubblicazione, è possibile caricare i gruppi di risorse database mediante il pacchetto Loader di WebSphere Commerce.

La prima parte di questo capitolo illustra i gruppi di risorse database di WebSphere Commerce e il modo in cui viene determinato un gruppo. La seconda parte descrive il processo di caricamento di questi gruppi di risorse nel database WebSphere Commerce. Prima di leggere questa sezione, è necessario rivedere le informazioni riportate in Capitolo 27, "Panoramica sul caricamento dei dati" a pagina 229, sul caricamento dei gruppi di risorse database con il pacchetto Loader.

Gruppi di risorse database

Le risorse database sono suddivise in gruppi per semplificare i processi di creazione e caricamento. Tali *gruppi di risorse database* comprendono un insieme di tabelle collegato logicamente. L'ordine in cui viene organizzato un gruppo di risorse database è importante perché i dati devono esistere prima di poter caricare la relazione tra dati.

Per caricare l'intero insieme di risorse database per un negozio, è necessario attenersi alla "Sequenza di caricamento delle risorse database". Per caricare un singolo gruppo di risorse database, è necessario assicurarsi che questo gruppo sia logicamente completo. Ad esempio, quando viene pubblicato un archivio negozio, è possibile omettere le risorse database del catalogo per pubblicarle in un secondo momento. In questo caso, le risorse database che dipendono dal catalogo (inventario, listini prezzi e alcuni dati relativi a spedizione e tassazione) restano non pubblicati. Per pubblicare i dati omessi, assicurarsi che le risorse database di catalogo siano logicamente complete: devono contenere articoli di base, voci di catalogo, attributi, ecc. E' necessario anche pubblicare le risorse database dipendenti ed esse devono essere logicamente complete. In altre parole, per ciascuna SKU devono essere definiti inventario, prezzo, spedizione e tasse. In questo caso, i dati di catalogo correlati logicamente completi vengono collettivamente detti *gruppo di risorse database del catalogo*.

Le risorse database di WebSphere Commerce descritte nei capitoli precedenti di questa guida possono essere organizzate in gruppi. Un gruppo è un insieme logicamente completo di dati che è possibile caricare singolarmente. Ciascun gruppo di risorse database è formato da tabelle di database WebSphere Commerce e ha dipendenze esterne, come descritto in Appendice E, "Gruppi di risorse database" a pagina 331. L'elenco di tabelle è basato sui negozi di esempio di WebSphere Commerce, ma è applicabile a qualsiasi negozio generico. Ricordare che l'elenco di tabelle relativo a ciascun gruppo di risorse database non è esauriente, ma fornisce una panoramica generale. Può essere necessario includere o escludere alcune tabelle a seconda di particolari esigenze.

Sequenza di caricamento delle risorse database

Per caricare correttamente i gruppi di risorse database, è necessario seguire un certo ordine. Ciascun gruppo è considerato strutturalmente completo e indipendente dagli altri gruppi. Esistono, tuttavia, relazioni di chiavi esterne

all'interno di ciascun gruppo di risorse database. Tali relazioni (con dati di altri gruppi) sono definiti *dipendenze esterne* di un gruppo di risorse database.

E' necessario soddisfare le dipendenze esterne di un gruppo di risorse database prima di caricarlo nel database WebSphere Commerce. Ciascun gruppo definito come dipendenza esterna di un dato gruppo di risorse database deve essere caricato per primo. E' possibile trovare l'elenco di dipendenze esterne e le tabelle correlate in "Dipendenze di gruppi di risorse database" a pagina 331.

Nota: Un negozio WebSphere Commerce richiede un proprietario. E' possibile utilizzare l'organizzazione predefinita, disponibile come proprietario predefinito.  Per caricare questo gruppo, creare una nuova organizzazione invece di utilizzare quella predefinita.

Caricare i gruppi di risorse database nel seguente ordine:

1. Gruppi di risorse database dipendenti solo dai dati di bootstrap.
 - a. Caricare prima le risorse database dell'**organizzazione**.
2. Gruppo di risorse database dipendente dal proprietario di evasione ordini.
 - a. Risorse database di **evasione ordini**. Eccetto il gruppo di risorse database dell'organizzazione, molti gruppi di risorse database hanno una dipendenza diretta o indiretta dai dati definiti in questo gruppo.
3. Gruppi di risorse database dipendenti dall'organizzazione proprietaria del negozio.
 - a. **Le risorse database di controllo accessi** dipendono dall'organizzazione proprietaria del negozio (ORGENTITY_ID). Nessuno degli altri gruppi di risorse database dipende dai dati definiti in questo gruppo, pertanto le risorse database di controllo accessi possono essere caricate in qualsiasi momento. Tuttavia, il proprietario del controllo accessi deve essere il proprietario del negozio.
 - b. Le risorse database del **negozio** dipendono dall'organizzazione proprietaria del negozio (ORGENTITY_ID).

Il negozio può fare riferimento a un centro di evasione ordini. L'organizzazione proprietaria del negozio può essere anche l'organizzazione proprietaria del centro di evasione ordini.

4. Gruppi di risorse database dipendenti dalle risorse del negozio. I gruppi di seguito riportati possono essere caricati in qualsiasi ordine:
 - a. Risorse database di **campagne**.
 - b. Risorse database di **comandi**.
 - c. Risorse database di **valute**.
 - d. Risorse database di **politiche**.
 - e. Risorse database di **spedizione**.
 - f. Risorse database di **tasse**.
5. Altri gruppi di risorse database.
 - a. Le risorse database di **catalogo** dipendono dai gruppi di risorse database di spedizione e tasse.
 - b. Le risorse database **predefinite del negozio** hanno dipendenze esterne dal gruppo di risorse database di spedizione. Se le risorse database di spedizione non esistono, non è necessario riempire questo gruppo.
 - c. Le risorse database di **contratti** dipendono dalle risorse dell'organizzazione. Le risorse database dei contratti non vengono caricate direttamente. Per

ulteriori informazioni, fare riferimento a “Pubblicazione delle risorse contratto” a pagina 287. E’ necessario caricare le risorse contratto dopo i gruppi di risorse database.

Fare riferimento a Appendice E, “Gruppi di risorse database” a pagina 331 per vedere il contenuto dei gruppi di risorse database formati dai negozi di esempio di WebSphere Commerce.

Caricamento di un negozio

Per facilitare il caricamento delle risorse database, sono disponibili pacchetti di esempio del sito Web di WebSphere Commerce. Tali pacchetti si basano sui negozi di esempio di WebSphere Commerce e contengono i file trovati nelle operazioni sopra riportate. E’ possibile scaricare i pacchetti dal seguente sito:

-  http://www.ibm.com/software/webservers/commerce/wc_be/downloads.html
-  http://www.ibm.com/software/webservers/commerce/wc_pe/downloads.html

Per caricare i dati XML relativi a un intero negozio nel database WebSphere Commerce, procedere come segue:

1. Rivedere le informazioni di seguito riportate:
 - a. Appendice B, “Creazione di dati” a pagina 319
 - b. Appendice E, “Gruppi di risorse database” a pagina 331, perché è necessario sapere quali file di risorse database WebSphere Commerce e quali tabelle di database sono coinvolte.
 - c. Capitolo 27, “Panoramica sul caricamento dei dati” a pagina 229, che fornisce le informazioni di sfondo per il pacchetto Loader.
2. Pianificare il processo di caricamento per un insieme completo di risorse database del negozio. Che si desideri caricare un singolo gruppo di risorse database come indicato in “Caricamento dei gruppi di risorse database” a pagina 279 o un intero negozio il processo di base resta invariato. Nei passi successivi, occorre utilizzare o creare i seguenti file:
 - a. Uno o più file di risorse database per ciascun gruppo. Quando si carica un negozio completo, sono necessari tutti i file di risorse database creati. Ad esempio, è necessario un file `.xml` di *risorse database* (come in `campaign.xml`, `catalog.xml` o `currency.xml`), e file `.xml` di *risorse database* separati, specifici della locale per la locale supportata dal negozio. Esempi di tali file sono forniti con i negozi di esempio di WebSphere Commerce nella seguente directory:

-  `unità:\WebSphere\CommerceServer\samplstores\
nome negozio di esempio\data\`
-  `unità:\Program
Files\WebSphere\CommerceServer\samplstores\
nome negozio di esempio\data\`
-  `/usr/WebSphere/CommerceServer/samplstores/
nome negozio di esempio/data/`
-   `/opt/WebSphere/CommerceServer/samplstores/
nome negozio di esempio/data/`
-  `/QIBM/ProdData/WebCommerce/samplstores/
nome negozio di esempio/data/`

Tener presente che non tutti i gruppi di risorse database richiedono informazioni specifiche della locale.

- b. Un nuovo file XML, che consolida tutti i file XML di risorse database del negozio, contiene i riferimenti entità XML e l'elemento root dell'intero negozio. Questo file viene definito *file XML di gruppo di risorse database principale* ed è possibile trovarlo nel pacchetto di esempio con il nome `nomestore-data-assets.xml`.
- c. Un nuovo file DTD che definisce tutti i tipi di dati richiesti dai file XML di un gruppo di risorse database definito come *file DTD di gruppo di risorse database principale*. E' possibile trovare tale file nel pacchetto di esempio con il nome `store-data-assets.dtd`.
- d. Un secondo file DTD che definisce le dipendenze esterne. Potrebbe essere necessario includere questo file nel *file DTD di gruppo di risorse database principale* ed è possibile trovarlo nel pacchetto di esempio con il nome `ForeignKeys.dtd`.
- e. Un terzo file DTD contenente la definizione di tutte le tabelle di WebSphere Commerce. Il file `wcs.dtd` esiste già in WebSphere Commerce e risiede nella seguente directory:

-  `unità:\WebSphere\CommerceServer\schema\xml\`
-  `unità:\Program Files\WebSphere\CommerceServer\schema\xml\`
-  `/usr/WebSphere/CommerceServer/schema/xml/`
-   `/opt/WebSphere/CommerceServer/schema/xml/`
-  `/QIBM/ProdData/WebCommerce/schema/xml/`

Potrebbe essere necessario includere questo file nel *file DTD di gruppo di risorse database principale* Se lo schema di WebSphere Commerce non è stato personalizzato, è possibile utilizzare questo file senza modifiche.

3. Creare i file XML di risorse database come descritto nei precedenti capitoli di questa guida. Se sono state completate le attività descritte nei capitoli precedenti, questi file XML esistono già. I file di risorse database non devono contenere dichiarazioni DTD o istruzioni sulle pagine all'inizio del file perché ciò potrebbe causare conflitti in fase di concatenamento. Inoltre, per semplicità, è possibile non creare gli elementi root. L'unico file che deve avere un elemento root è il file XML di gruppo di risorse database principale.

Nota: Se si dispone di file di risorse database per più lingue, ogni file deve cominciare con `<?xml encoding= codifica specifica della locale>`. Ad esempio, i file di risorse database per l'inglese devono specificare `<?xml encoding = "UTF-8"?>`, quelli per il francese devono specificare invece `<?xml encoding = "ISO-8859-1"?>`.

4. Creare il file XML di risorse database principale per l'intero insieme di dati del negozio. Questo file contiene le entità riferimento per includere i diversi file XML di risorse database del negozio. Utilizzando entità riferimento esterne, è possibile concatenare i file XML allo scopo di semplificare il comando ID Resolve e il processo di caricamento. Inoltre, gli alias interni utilizzati in ciascun file XML possono essere esterni per un altro file XML di risorse database di un gruppo, o di vari gruppi se si caricano più gruppi contemporaneamente. Un parser XML sostituisce il contenuto del file a cui fa riferimento l'entità riferimento esterna invece del riferimento esterno.

Utilizzando il seguente esempio per il caricamento dell'intero insieme di dati del negozio, è possibile creare il file di gruppo di risorse database in base a questo estratto:

```

<?xml version="1.0"?>
<!DOCTYPE import SYSTEM "all-store-assets.dtd">
<import>
<!-- Fulfillment data group -->
&fulfillment.xml;

<!-- Store data group -->
&store.xml;
&en_US_store.xml;
&fr_FR_store.xml;

<!-- Tax data group -->
&tax.xml;
&en_US_tax.xml;
&fr_FR_tax.xml;
&taxfulfill.xml;

<!-- Shipping data group -->
&shipping.xml;
&en_US_shipping.xml;
&fr_FR_shipping.xml;
&shipfulfill.xml;

<!-- Catalog data group -->
&catalog.xml;
&en_US_catalog.xml;
&fr_FR_catalog.xml;
&storecatalog.xml;
&storefulfill.xml;
&offering.xml;
&store-catalog-tax.xml;
&store-catalog-shipping.xml;

<!-- Currency data group -->
&currency.xml;
&en_US_currency.xml;
&fr_FR_currency.xml;

<!-- Campaign data group -->
&campaign.xml;
&en_US_campaign.xml;
&fr_FR_campaign.xml;

<!-- Business policy data group -->
&businesspolicy.xml;
&en_US_businesspolicy.xml;
&fr_FR_businesspolicy.xml;

<!-- Access control data group -->
&accesscontrol.xml;
&en_US_accesscontrol.xml;
&fr_FR_accesscontrol.xml;

<!-- Other data groups -->
&command.xml;
&store-default.xml;
</import>

```

dove

- import è l'elemento root del documento XML. L'elemento root è stato già definito nel file wcs.dtd, fornito con WebSphere Commerce, e include le definizioni per tutte le tabelle del database WebSphere Commerce. Tuttavia, se è stato personalizzato lo schema di WebSphere Commerce, è possibile

utilizzare un diverso elemento root. E' possibile generare un nuovo file DTD che riflette lo schema personalizzato oppure aggiornare il file wcs.dtd esistente.

- all-store-assets.dtd si riferisce al nome del file DTD di gruppo di risorse database principale creato nell'operazione successiva.
- Il testo commentato distingue i diversi gruppi di risorse database del negozio.
- &risorsa database.xml; è un riferimento entità XML per il file XML di risorse database. Il percorso e la posizione sono definiti nel file DTD di gruppo di risorse database. Tale nome verrà modificato in modo da corrispondere ai file di risorse database già creati per ciascun gruppo.
- &locale_risorsa database.xml; è richiesto per ciascuna lingua supportata dal negozio. Se il negozio supporta una sola lingua, è richiesto il riferimento a un solo file. Se il negozio supporta più lingue, è richiesto un riferimento per ciascuna lingua. L'estratto sopra riportato presuppone che il negozio supporti le lingue inglese e francese.

5. Creare un file DTD di gruppo risorse database principale che definisca le entità sopra riportate e gli altri file DTD richiesti dalle risorse database.

Utilizzando il seguente esempio per l'intero insieme di risorse database del negozio come guida, è possibile creare il file DTD di gruppo di risorse database principale:

```
<!ENTITY % wcs.dtd SYSTEM "percorso assoluto per il file di WebSphere Commerce wcs.dtd">
%wcs.dtd;

<!ENTITY % NonStoreForeignKeys.dtd SYSTEM "NonStoreForeignKeys.dtd">
%NonStoreForeignKeys.dtd;
<!ENTITY fulfillment.xml SYSTEM "data/fulfillment.xml">
<!ENTITY en_US_fulfillment.xml SYSTEM "data/en_US/fulfillment.xml">
<!ENTITY fr_FR_fulfillment.xml SYSTEM "data/fr_FR/fulfillment.xml">

<!ENTITY store.xml SYSTEM "data/store.xml">
<!ENTITY en_US_store.xml SYSTEM "data/en_US/store.xml">
<!ENTITY fr_FR_store.xml SYSTEM "data/fr_FR/store.xml">

<!ENTITY tax.xml SYSTEM "data/tax.xml">
<!ENTITY en_US_tax.xml SYSTEM "data/en_US/tax.xml">
<!ENTITY fr_FR_tax.xml SYSTEM "data/fr_FR/tax.xml">
<!ENTITY taxfulfill.xml SYSTEM "data/taxfulfill.xml">

<!ENTITY shipping.xml SYSTEM "data/shipping.xml">
<!ENTITY en_US_shipping.xml SYSTEM "data/en_US/shipping.xml">
<!ENTITY fr_FR_shipping.xml SYSTEM "data/fr_FR/shipping.xml">
<!ENTITY shipfulfill.xml SYSTEM "data/shipfulfill.xml">

<!ENTITY catalog.xml SYSTEM "data/catalog.xml">
<!ENTITY en_US_catalog.xml SYSTEM "data/en_US/catalog.xml">
<!ENTITY fr_FR_catalog.xml SYSTEM "data/fr_FR/catalog.xml">
<!ENTITY store-catalog.xml SYSTEM "data/store-catalog.xml">
<!ENTITY storefulfill.xml SYSTEM "data/storefulfill.xml">
<!ENTITY offering.xml SYSTEM "data/offering.xml">
<!ENTITY store-catalog-tax.xml SYSTEM "data/store-catalog-tax.xml">
<!ENTITY store-catalog-shipping.xml SYSTEM "data/store-catalog-shipping.xml">

<!ENTITY currency.xml SYSTEM "data/currency.xml">
<!ENTITY en_US_currency.xml SYSTEM "data/en_US/currency.xml">
<!ENTITY fr_FR_currency.xml SYSTEM "data/fr_FR/currency.xml">

<!ENTITY campaign.xml SYSTEM "data/campaign.xml">
<!ENTITY en_US_campaign.xml SYSTEM "data/en_US/campaign.xml">
<!ENTITY fr_FR_campaign.xml SYSTEM "data/fr_FR/campaign.xml">
```

```

<!ENTITY businesspolicy.xml SYSTEM "data/businesspolicy.xml">
<!ENTITY en_US_businesspolicy.xml SYSTEM "data/en_US/businesspolicy.xml">
<!ENTITY fr_FR_businesspolicy.xml SYSTEM "data/fr_FR/businesspolicy.xml">

<!ENTITY accesscontrol.xml SYSTEM "data/accesscontrol.xml">
<!ENTITY en_US_accesscontrol.xml SYSTEM "data/en_US/accesscontrol.xml">
<!ENTITY fr_FR_accesscontrol.xml SYSTEM "data/fr_FR/accesscontrol.xml">

<!ENTITY command.xml SYSTEM "data/command.xml">
<!ENTITY store-defaults.xml SYSTEM "data/store-defaults.xml">

```

dove

- `wcs.dtd` si riferisce al file DTD contenente i dati definiti al di fuori del gruppo di risorse database. Questo file, fornito con WebSphere Commerce, definisce anche l'elemento root utilizzato nel file XML di gruppo di risorse database.
- `NonStoreForeignKeys.dtd` si riferisce al file DTD che definisce gli elementi non root. Questo file contiene tutte le definizioni e dichiarazioni di riferimento a entità XML per le dipendenze esterne al di fuori del gruppo di risorse database. I file XML hanno riferimenti a valori di chiavi esterne non creati come parte del gruppo di risorse database e devono già essere caricati nel database prima di tale gruppo.

Nota: Assicurarsi che il percorso sia identificato correttamente. In questo esempio, il file si trova nella stessa directory del file DTD di gruppo di risorse database principale.

- `store.xml`, `en_US_store.xml` e `fr_FR_store.xml` sono le entità di riferimento esterne utilizzate nel file XML di gruppo di risorse database supponendo che il negozio supporti le lingue inglese e francese. Per utilizzare il riferimento, seguire la convenzione dei riferimenti entità *&nome_alias*;
- `risorsa database.xml` si riferisce al nome dei file da cui vengono caricate le risorse database. Tale nome verrà modificato in modo da corrispondere ai file di risorse database già creati per ciascun gruppo. Tener presente che i file XML specifici delle locale si trovano nella seguente directory:

-  *unità*: \WebSphere\CommerceServer\samplstores\
nome negozio esempio\data/locale
-  *unità*: \Program
Files\WebSphere\CommerceServer\samplstores\
nome negozio esempio\data/locale
-  /usr/WebSphere/CommerceServer/samplstores/
nome negozio esempio/data/locale/
-   /opt/WebSphere/CommerceServer/samplstores/
nome negozio esempio/data/locale/
-  /QIBM/ProdData/WebCommerce/samplstores/
nome negozio esempio/data/locale/

- i file `percorso_risorse database.xml` sono richiesti per ciascuna lingua supportata dal negozio e si trovano nelle directory sopra riportate. Se il negozio supporta una sola lingua, è richiesto il riferimento a un solo file. Se il negozio supporta più lingue, è richiesto un file specifico della locale per ciascuna lingua. L'estratto sopra riportato presuppone che il negozio supporti le lingue inglese e francese.
6. Ciascun gruppo di risorse database richiede informazioni definite al di fuori del dominio o dell'insieme di dati, perché ciascun gruppo può avere

dipendenze esterne. E' possibile fornire questi dati in un file DTD. Ad esempio, il gruppo di risorse database del negozio ha le seguenti dipendenze esterne:

```
bootstrap.LANGUAGE.LANGUAGE_ID, bootstrap.MEMBER.MEMBER_ID,  
bootstrap.SETCURR.SETCURR_ID, fulfillment.FFMCENTER.FFMCENTER_ID
```

Durante il caricamento di un gruppo di risorse database o dell'intero insieme di risorse del negozio, è necessario definire le dipendenze esterne dal database WebSphere Commerce. Per utilizzare questi dati, seguire il riferimento entità XML corrispondente. Ad esempio, per utilizzare i dati definiti dall'entità `ffmcenter_id`, scrivere `&ffmcenter_id`; nel file XML. Utilizzando l'esempio per le risorse database di seguito riportato come guida, è possibile creare il file DTD basato su questo estratto, `NonStoreForeignKeys.dtd`:

```
<!ENTITY en_US "-1">  
<!ENTITY fr_FR "-2">  
<!ENTITY de_DE "-3">  
<!ENTITY it_IT "-4">  
<!ENTITY es_ES "-5">  
<!ENTITY pt_BR "-6">  
<!ENTITY zh_CN "-7">  
<!ENTITY zh_TW "-8">  
<!ENTITY ko_KR "-9">  
<!ENTITY ja_JP "-10">  
<!ENTITY MEMBER_ID "-2000">  
<!ENTITY ffmcenter_id "10001">
```

dove

- `MEMBER_ID` è il numero di riferimento interno che identifica il proprietario del negozio.
- `ffmcenter` è il numero di riferimento per il centro di evasione ordini del negozio. Poiché il negozio può utilizzare più centri di evasione ordini, è possibile definirne più d'uno nel file `NonStoreForeignKeys.dtd`.
- *locale* è il numero di riferimento WebSphere Commerce per ciascuna locale (identificata dalla lingua nazionale). I valori risiedono nella tabella di database `LANGUAGE`.

Nota: Se si suddivide un archivio negozio esistente in gruppi di risorse database, assicurarsi che tutti i riferimenti, ad esempio quelli all'alias `@ffmcenter_id`, siano sostituiti con il riferimento all'entità corrispondente: `&ffmcenter_id`;

7. Una volta creati tutti i file di dati, eseguire il comando `IDResolve` sul file XML di gruppo risorse database principale per risolvere i dati come indicato in "Comando ID Resolve" a pagina 234.
8. Eseguire il comando `Load` sul file di dati risolto, come indicato in "Comando Load" a pagina 242. Per verificare il processo di caricamento, fare riferimento ai file di log:

-  `idresgen.db2.log` e `massload.db2.log`
-  `idresgen.oracle.log` e `massload.oracle.log`

I file di log risiedono in:

-  `unitâ:\WebSphere\CommerceServer\logs\`
-  `unitâ:\Program Files\WebSphere\CommerceServer\logs\`
-  `/usr/WebSphere/CommerceServer/logs/`

-   /opt/WebSphere/CommerceServer/logs/
 -  /QIBM/ProdData/WebCommerce/logs/
9.  Eseguire il comando AccountImport come indicato in “Pubblicazione di risorse conto commerciale” a pagina 286.
 10. Se applicabile, pubblicare i contratti come indicato in “Pubblicazione delle risorse contratto” a pagina 287.
 11. Completare le attività riportate in “Pubblicazione dei file di configurazione e delle risorse facciata del negozio mediante copia su WebSphere Commerce Server” a pagina 290

Caricamento dei gruppi di risorse database

Per facilitare il caricamento delle risorse database, sul sito Web di WebSphere Commerce sono disponibili pacchetti di esempio. Tali pacchetti si basano sui negozi di esempio di WebSphere Commerce e contengono i file trovati nelle operazioni sopra riportate. E' possibile scaricare i pacchetti dal seguente sito:

-  http://www.ibm.com/software/webservers/commerce/wc_be/downloads.html
-  http://www.ibm.com/software/webservers/commerce/wc_pe/downloads.html

Per caricare i dati XML relativi a un singolo gruppo di risorse database nel database WebSphere Commerce, procedere come segue:

1. Rivedere le informazioni di seguito riportate:
 - a. Appendice B, “Creazione di dati” a pagina 319
 - b. Appendice E, “Gruppi di risorse database” a pagina 331, perché è necessario sapere quali file di risorse WebSphere Commerce e quali tabelle di database sono coinvolte.
 - c. Capitolo 27, “Panoramica sul caricamento dei dati” a pagina 229, che fornisce le informazioni di sfondo per il pacchetto Loader.
2. Pianificare il processo di caricamento e decidere quale gruppo di risorse database caricare. Che si desideri caricare l'intero insieme di risorse database del negozio come indicato in “Caricamento di un negozio” a pagina 273 o un singolo gruppo di risorse database, il processo di base è lo stesso. Nei passi successivi, occorre utilizzare o creare i seguenti file:
 - a. Uno o più file di risorse database, a seconda del gruppo scelto. Ad esempio, per caricare le risorse gruppo database del negozio, sono necessari un file store.xml e un file store.xml separato per ciascuna locale supportata dal negozio. Esempi di tali file sono forniti con i negozi di esempio di WebSphere Commerce nella seguente directory:
 -  *unità*:\WebSphere\CommerceServer\samplstores\
nome negozio di esempio\data
 -  *unità*:\Program
Files\WebSphere\CommerceServer\samplstores\
nome negozio di esempio\data
 -  /usr/WebSphere/CommerceServer/samplstores/
nome negozio di esempio/data
 -   /opt/WebSphere/CommerceServer/samplstores/
nome negozio di esempio/data

-  /QIBM/ProdData/WebCommerce/samplestores/
nome negozio di esempio/data

Tener presente che non tutti i gruppi di risorse database richiedono informazioni specifiche della locale.

- Un nuovo file XML, che consolida tutti i file di risorse database XML e contiene i riferimenti entità XML e l'elemento root delle risorse database. Questo file viene definito *file XML di gruppo di risorse database principale*. E' possibile trovare tale file nel pacchetto di esempio con il nome, store-all-assets.xml.
- Un nuovo file DTD che definisce tutti i tipi di dati richiesti dai file XML di un gruppo di risorse database definito come *file DTD di gruppo di risorse database principale*. E' possibile trovare tale file nel pacchetto di esempio con il nome store-all-assets.dtd.
- Un secondo file DTD che definisce le dipendenze esterne. Potrebbe essere necessario includere questo file nel *file DTD di gruppo di risorse database principale* ed è possibile trovarlo nel pacchetto di esempio con il nome ForeignKeys.dtd.
- Un terzo file DTD contenente la definizione di tutte le tabelle di WebSphere Commerce. Il file wcs.dtd esiste già in WebSphere Commerce e risiede nella seguente directory:

-  unità:\WebSphere\CommerceServer\schema\xml\
-  unità:\Program Files\WebSphere\CommerceServer\schema\xml\
-  /usr/WebSphere/CommerceServer/schema/xml/
-   /opt/WebSphere/CommerceServer/schema/xml/
-  /QIBM/ProdData/WebCommerce/schema/xml/

Potrebbe essere necessario includere questo file nel *file DTD di gruppo di risorse database principale*. Se lo schema di WebSphere Commerce non è stato personalizzato, è possibile utilizzare questo file senza modifiche.

- Creare i file XML di risorse database per il gruppo come descritto nei precedenti capitoli di questa guida. Se sono state completate le attività descritte nei capitoli precedenti, questi file XML esistono già. I file di risorse database non devono contenere dichiarazioni DTD o istruzioni sulle pagine all'inizio del file perché ciò potrebbe causare conflitti in fase di concatenamento. Inoltre, per semplicità, è possibile non creare gli elementi root. L'unico file che deve avere un elemento root è il file XML di gruppo di risorse database principale.

Nota: Se si dispone di file di risorse database per più lingue, ogni file deve cominciare con `<?xml encoding = codifica specifica della locale>`. Ad esempio, i file di risorse database per l'inglese devono specificare `<?xml encoding = "UTF-8"?>`, quelli per il francese devono specificare invece `<?xml encoding = "ISO-8859-1"?>`.

- Creare il file XML di gruppo di risorse database principale per ciascun gruppo da caricare. Questo file contiene le entità di riferimento per includere vari file XML in uno o più gruppi di risorse database. Utilizzando entità riferimento esterne, è possibile concatenare i file XML allo scopo di semplificare il comando ID Resolve e il processo di caricamento. Inoltre, gli alias interni utilizzati all'interno di ciascun file XML possono essere esterni per un altro file di dati XML all'interno di un gruppo (o di più gruppi, se si caricano più

gruppi contemporaneamente). Un parser XML sostituisce il contenuto del file a cui fa riferimento l'entità riferimento esterna invece del riferimento esterno.

Utilizzando il seguente esempio per il caricamento del singolo gruppo di risorse database del negozio come guida, è possibile creare il file XML di gruppo di risorse database in base a questo estratto:

```
<?xml version="1.0"?>
<!DOCTYPE import SYSTEM "store-assets.dtd">
<import>
&store.xml;
&en_US_store.xml;
&fr_FR_store.xml;
</import>
```

dove

- `import` è l'elemento root del documento XML. L'elemento root è stato già definito nel file `wcs.dtd`, fornito con WebSphere Commerce, e include le definizioni per tutte le tabelle del database WebSphere Commerce. Tuttavia, se è stato personalizzato lo schema di WebSphere Commerce, è possibile utilizzare un diverso elemento root. E' possibile generare un nuovo file DTD che riflette lo schema personalizzato oppure aggiornare il file `wcs.dtd`.
 - `store-assets.dtd` si riferisce al nome del file DTD di gruppo risorse database principale creato al passo successivo.
 - `&store.xml`; è un riferimento entità XML per il file XML di gruppo di risorse database. Il percorso e la posizione sono definiti nel file DTD di gruppo di risorse database. Tale nome verrà modificato in modo da corrispondere ai file di risorse già creati per ciascun gruppo.
 - `locale_store.xml`; è richiesto per ciascuna lingua supportata dal negozio. Se il negozio supporta una sola lingua, è richiesto il riferimento a un solo file. Se il negozio supporta più lingue, è richiesto un riferimento per ciascuna lingua. L'estratto sopra riportato presuppone che il negozio supporti le lingue inglese e francese.
5. Creare un file DTD di gruppo risorse database principale che definisca le entità sopra riportate e gli altri file DTD richiesti dal gruppo.

Utilizzando il seguente esempio per il caricamento di un singolo gruppo di dati del negozio come guida, è possibile creare il file DTD di gruppo di risorse database principale per qualsiasi gruppo di dati:

```
<!ENTITY % wcs.dtd SYSTEM "percorso assoluto per il file di WebSphere Commerce wcs.dtd">
%wcs.dtd;

<!ENTITY % NonStoreForeignKeys.dtd SYSTEM "NonStoreForeignKeys.dtd">
%NonStoreForeignKeys.dtd;
<!ENTITY store.xml SYSTEM "store.xml">
<!ENTITY en_US_store.xml SYSTEM "en_US/store.xml">
<!ENTITY fr_FR_store.xml SYSTEM "fr_FR/store.xml">
```

dove

- `wcs.dtd` si riferisce al file DTD contenente i dati definiti al di fuori del gruppo di risorse database. Questo file, fornito con WebSphere Commerce, risolve e definisce anche l'elemento root utilizzato nel file XML di gruppo di risorse database.
- `NonStoreForeignKeys.dtd` si riferisce al file DTD che definisce gli elementi non root. Questo file contiene tutte le definizioni e dichiarazioni di riferimento a entità XML per le dipendenze esterne al di fuori del gruppo di risorse database. I file XML hanno riferimenti a valori di chiavi esterne non creati come parte del gruppo di risorse database e devono già essere caricati nel database prima di tale gruppo.

Nota: Assicurarsi che il percorso sia identificato correttamente. In questo esempio, il file si trova nella stessa directory del file DTD di gruppo di risorse database.

- store.xml, en_US_store.xml e fr_FR_store.xml sono le entità di riferimento esterne utilizzate nel file XML di gruppo di risorse database. Per utilizzare il riferimento, seguire la convenzione dei riferimenti entità *&nome_alias;*.
- store.xml si riferisce al file di dati per il gruppo dal quale vengono caricate le risorse database. Tale nome verrà modificato in modo da corrispondere ai file di risorse database già creati per ciascun gruppo. Tener presente che i file XML specifici delle locale si trovano nella seguente directory:

- **NT** *unità:*\WebSphere\CommerceServer\samplstores\
nome negozio esempio\data/locale
- **2000** *unità:*\Program
Files\WebSphere\CommerceServer\samplstores\
nome negozio esempio\data/locale
- **AIX** /usr/WebSphere/CommerceServer/samplstores/
nome negozio esempio/data/locale/
- **Solaris** **Linux** /opt/WebSphere/CommerceServer/samplstores/
nome negozio esempio/data/locale/
- **400** /QIBM/ProdData/WebCommerce/samplstores/
nome negozio esempio/data/locale/

- I file *percorso_store.xml* sono richiesti per ciascuna lingua supportata dal negozio e si trovano nelle directory sopra riportate. Se il negozio supporta una sola lingua, è richiesto il riferimento a un solo file. Se il negozio supporta più lingue, è richiesto un file specifico della locale per ciascuna lingua. L'estratto sopra riportato presuppone che il negozio supporti le lingue inglese e francese.

6. Ciascun gruppo di risorse database richiede informazioni definite al di fuori del dominio o dell'insieme di dati, perché ciascun gruppo può avere dipendenze esterne. E' possibile fornire questi dati in un file DTD. Ad esempio, il gruppo di risorse database del negozio ha le seguenti dipendenze esterne:

```
bootstrap.LANGUAGE.LANGUAGE_ID, bootstrap.MEMBER.MEMBER_ID,  
bootstrap.SETCURR.SETCURR_ID, fulfillment.FFMCENTER.FFMCENTER_ID
```

Durante il caricamento di un gruppo o dell'intero insieme di dati del negozio, è necessario definire le seguenti dipendenze del database WebSphere Commerce. Per utilizzare questi dati, seguire il riferimento entità XML corrispondente. Ad esempio, per utilizzare i dati definiti dall'entità *ffmcenter_id*, scrivere *&ffmcenter_id*; nel file XML. Utilizzando l'esempio per il gruppo di risorse database del negozio di seguito riportato come guida, è possibile creare il file DTD basato su questo estratto, *NonStoreForeignKeys.dtd*:

```
<!ENTITY en_US "-1">  
<!ENTITY fr_FR "-2">  
<!ENTITY de_DE "-3">  
<!ENTITY it_IT "-4">  
<!ENTITY es_ES "-5">  
<!ENTITY pt_BR "-6">  
<!ENTITY zh_CN "-7">  
<!ENTITY zh_TW "-8">
```

```
<!ENTITY ko_KR "-9">
<!ENTITY ja_JP "-10">
<!ENTITY MEMBER_ID "-2000">
<!ENTITY ffmcenter_id "10001">
```

dove

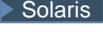
- MEMBER_ID è il numero di riferimento interno che identifica il proprietario del negozio.
- ffmcenter è il numero di riferimento per il centro di evasione ordini del negozio. Poiché il negozio può utilizzare più centri di evasione ordini, è possibile definirne più d'uno nel file NonStoreForeignKeys.dtd.
- locale è il numero di riferimento WebSphere Commerce per ciascuna locale (identificata dalla lingua nazionale). I valori risiedono nella tabella di database LANGUAGE.

Nota: Se si suddivide un archivio negozio esistente in gruppi di risorse database, assicurarsi che tutti i riferimenti, ad esempio quelli all'alias @ffmcenter_id, siano sostituiti con il riferimento all'entità corrispondente: &ffmcenter_id;.

7. Una volta creati tutti i file di dati, eseguire il comando IDResolve sul file XML di gruppo risorse database per risolvere i dati come indicato in "Comando ID Resolve" a pagina 234.
8. Eseguire il comando Load sul file di dati risolto, come indicato in "Comando Load" a pagina 242. Per verificare il processo di caricamento, fare riferimento ai file di log:

-  idresgen.db2.log e massload.db2.log
-  idresgen.oracle.log e massload.oracle.log

I file di log risiedono in:

-  *unità:* \WebSphere\CommerceServer\logs\
-  *unità:* \Program Files\WebSphere\CommerceServer\logs\
-  /usr/WebSphere/CommerceServer/logs/
-   /opt/WebSphere/CommerceServer/logs/
-  /QIBM/ProdData/WebCommerce/logs/

9.  Eseguire il comando AccountImport come indicato in "Pubblicazione di risorse conto commerciale" a pagina 286.
10. Se applicabile, pubblicare i contratti come indicato in "Pubblicazione delle risorse contratto" a pagina 287.
11. Completare le attività riportate in "Pubblicazione dei file di configurazione e delle risorse facciata del negozio mediante copia su WebSphere Commerce Server" a pagina 290

Capitolo 29. Pubblicazione dei contratti e dei conti commerciali

Alcune delle risorse database del negozio (conti commerciali e contratti) non possono essere caricate con il pacchetto Loader. E' possibile pubblicare queste risorse database mediante Servizi del negozio o la riga comandi come parte dell'opzione di pubblicazione di un intero negozio descritta in Capitolo 26, "Pubblicazione di un negozio completo" a pagina 215, oppure è possibile pubblicare conti commerciali e contratti utilizzando i corrispondenti comandi. Tali comandi sono i seguenti:

- **AccountImport**— Crea conti dal file `businessaccount.xml` dell'archivio negozio.
- **ContractImportApprovedVersion**—Crea un contratto dal file `contract.xml`. Se il contratto è in stato attivo, il comando crea e distribuisce il contratto. Anche se il file `contract.xml` contiene più contratti, è sufficiente richiamare il comando una sola volta.
- **ProductSetPublish** — Sincronizza i dati dell'insieme di prodotti nelle tabelle di database di insiemi di prodotti con il catalogo prima della creazione di contratti e conti commerciali. Servizi del negozio e il programma di utilità per la pubblicazione della riga comandi richiamano il comando **ProductSetPublish** che a sua volta richiama i comandi **AccountImport** e **ContractImportApprovedVersion**.

Nota: Per ulteriori informazioni su questi comandi, consultare la guida in linea di WebSphere Commerce.

Le risorse conto commerciale sono incluse come file XML in alcuni degli archivi negozio forniti con WebSphere Commerce. Tuttavia, si consiglia di creare le risorse conto commerciale mediante gli strumenti forniti invece di creare i file XML per quelle risorse. Per ulteriori informazioni sulla creazione delle risorse con gli strumenti forniti, consultare la guida in linea di WebSphere Commerce. Le istruzioni per la pubblicazione di conti commerciali sono incluse nelle sezioni successive, in caso si scelga di pubblicare i corrispondenti file XML forniti con gli archivi di esempio o di crearne uno nuovo.

Nota: Se non si utilizza Servizi del negozio per pubblicare contratti o conti commerciali, è necessario pubblicare le risorse negozio e catalogo prima di conti e contratti. In particolare, sono necessari identificativi di catalogo e negozio oltre all'ID dell'organizzazione che possiede il negozio e agli ID delle organizzazioni acquirenti associate al contratto. Se i termini e le condizioni del contratto non specificano un particolare catalogo, non è necessario pubblicare un catalogo prima di un contratto o conto commerciale.

Se si pubblicano queste risorse mediante Servizi del negozio o la riga comandi, non dimenticare di selezionare l'opzione relativa al catalogo, se il negozio non ha già un catalogo pubblicato. Se si pubblicano queste risorse utilizzando i comandi corrispondenti, assicurarsi di aver già caricato le risorse sopra elencate nel database.

Publicazione di contratti e conti commerciali con Servizi del negozio o la riga comandi

E' possibile pubblicare contratti e conti commerciali mediante Servizi del negozio o utilizzando il programma di utilità per la pubblicazione dalla riga comandi. Per pubblicare contratti e conti commerciali mediante Servizi del negozio o la riga comandi, le risorse devono essere impacchettate in formato archivio negozio. Per ulteriori informazioni sull'impacchettamento delle risorse facciata di negozio come archivio negozio, consultare Parte 6, "Organizzazione del negozio" a pagina 205.

Con Servizi del negozio o con la riga comandi, è possibile scegliere di pubblicare tutti i tipi di risorse di un archivio negozio (incluso le risorse facciata negozio, le risorse dati del negozio e i bundle di risorse) oppure solo un tipo. Per istruzioni dettagliate sulla pubblicazione delle risorse con Servizi del negozio o con la riga comandi, consultare la guida in linea di WebSphere Commerce.

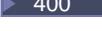
Publicazione di contratti e conti commerciali mediante i comandi

Se si preferisce non impacchettare le risorse come archivio negozio, è comunque possibile pubblicare contratti e conti commerciali utilizzando i comandi corrispondenti:

- **AccountImport**— Crea conti dal file `businessaccount.xml` dell'archivio negozio.
- **ContractImportApprovedVersion**— Importa un contratto approvato o attivo in WebSphere Commerce Server da un file XML. Prima di importarlo, il comando verifica che il contratto contenga le condizioni necessarie e che sia valido.
- **ProductSetPublish** — Sincronizza i dati dell'insieme di prodotti nelle tabelle di database di insiemi di prodotti con il catalogo prima della creazione di contratti e conti commerciali. Servizi del negozio e il programma di utilità per la pubblicazione della riga comandi richiamano il comando **ProductSetPublish** che a sua volta richiama i comandi **AccountImport** e **ContractImportApprovedVersion**.

Publicazione di risorse conto commerciale

Per pubblicare le risorse conto commerciale, procedere come segue:

1. Mediante la console di gestione, aggiornare il registro di visualizzazione o riavviare l'istanza di WebSphere Commerce. Per ulteriori informazioni, fare riferimento alla guida in linea di WebSphere Commerce.
2. Copiare `businessaccount.xml` nella seguente directory:
 -  `NT` unità:\WebSphere\CommerceServer\xml\trading
 -  `2000` unità:\Program Files\WebSphere\CommerceServer\xml\trading
 -  `AIX` /usr/WebSphere/CommereServer/xml/trading
 -  `Solaris` /opt/WebSphere/CommerceServer/xml/trading
 -  `Linux` /opt/WebSphere/CommerceServer/xml/trading
 -  `400` /QIBM/UserData/WebCommerce/instances/instancename/xml/trading
3. Aprire `businessaccount.xml` e apportare le seguenti modifiche:
 - Sostituire tutte le ricorrenze di `&STORE_IDENTIFIER`; con l'identificativo del negozio.
 - Sostituire tutte le ricorrenze di `&MEMBER_IDENTIFIER`; con il nome distinto di membro per il negozio.

Nota: Se si sta utilizzando un file `businessaccount.xml` che fa parte dell'archivio negozio creato con Servizi del negozio, questa operazione è già stata completata.

4. Salvare e chiudere il file.
5. Aprire la console di gestione. Collegarsi come responsabile.
6. In un browser, immettere:
 - `https://nomehost:8000/webapp/wcs/stores/servlet/AccountImport?fileName=businessaccount.xml&URL=URL a cui reindirizzarsi al completamento`

Nota: Per ulteriori informazioni sui parametri e la sintassi del comando, fare riferimento alla guida in linea di WebSphere Commerce.

Publicazione delle risorse contratto

Per pubblicare le risorse contratto, procedere come segue:

1. Copiare `contract.xml` nella seguente directory:

Nota: Questo percorso è configurabile. Il seguente formato è quello predefinito.

-  unità:\WebSphere\CommerceServer\xml\trading
 -  unità:\Program Files\WebSphere\CommerceServer\xml\trading
 -  /usr/WebSphere/CommerceServer/xml/trading
 -  /opt/WebSphere/CommerceServer/xml/trading
 -  /opt/WebSphere/CommerceServer/xml/trading
 -  /QIBM/UserData/WebCommerce/instances/*nomeistanza*/xml/trading
2. Aprire `contract.xml` e apportare le seguenti modifiche:
 - Sostituire tutte le ricorrenze di `&STORE_IDENTIFIER;` con l'identificativo del negozio.
 - Sostituire tutte le ricorrenze di `&MEMBER_IDENTIFIER;` con il nome distinto di membro per il negozio.
 3. Salvare e chiudere il file.
 4. Aprire la console di gestione. Collegarsi come responsabile.
 5. In un browser, immettere:
 - `https://nomehost:numeroporta/webapp/wcs/tools/servlet/ContractImportApprovedVersion?fileName=contract.xml&targetStoreId=store_id&URL=ContractDisplay`
 6. Se il negozio contiene più file `contract.xml` (ad esempio file di contratto specifici delle locale), ripetere le operazioni da 1 a 5 per ciascun file `contract.xml`.

Capitolo 30. Pubblicazione delle risorse della facciata e dei file di configurazione del negozio

La pubblicazione delle risorse della facciata del negozio, dei file HTML e JSP, dei file properties, dei bundle di risorse, delle immagini e dei grafici che costituiscono le pagine del negozio fa parte del processo di creazione di un negozio funzionale. E' possibile pubblicare le risorse facciata di negozio utilizzando Servizi del negozio o la riga comandi come parte della pubblicazione di un'opzione di negozio completo, come descritto in Capitolo 26, "Pubblicazione di un negozio completo" a pagina 215, oppure è possibile pubblicare le risorse facciata di negozio semplicemente copiandole da una percorso specificato sul WebSphere Commerce Server.

Se si pubblicano i file JSP contenuti nei negozi di esempio  ToolTech e NewFashion e si desidera configurare il negozio in modo da utilizzare le funzioni di collaborazione, sarà necessario pubblicare anche i file di configurazione del negozio che fanno parte di quell'archivio negozio. Entrambi i negozi contengono i seguenti file di configurazione del negozio:

- tools_properties.zip
- tools_xml.zip
- runtime_xml.zip

Se si pubblica un negozio completo mediante Servizi del negozio o la riga comandi (selezionando tutte le opzioni di pubblicazione), vengono pubblicati anche i file di configurazione. Tuttavia, se si sceglie di pubblicare le risorse facciata di negozio copiandole sul WebSphere Commerce Server, sarà necessario copiare anche i file di configurazione del negozio.

Pubblicazione delle risorse della facciata e dei file di configurazione del negozio mediante Servizi del negozio o la riga comandi

E' possibile pubblicare le risorse facciata e i file di configurazione del negozio mediante Servizi del negozio o utilizzando il programma di utilità per la pubblicazione dalla riga comandi.

Nota: Per pubblicare i file di configurazione del negozio con Servizi del negozio o il programma di utilità per la pubblicazione, è necessario pubblicare tutte le risorse del negozio, incluso del risorse Web e le risorse dati. Non è possibile pubblicare solo i file di configurazione del negozio.

Per pubblicare le risorse facciata del negozio e i file di configurazione con Servizi del negozio o con la riga comandi, è necessario impacchettare le une e gli altri nel formato archivio negozio. Per ulteriori informazioni sull'impacchettamento delle risorse facciata di negozio come archivio negozio, consultare Parte 6, "Organizzazione del negozio" a pagina 205.

Con Servizi del negozio o con la riga comandi, è possibile scegliere di pubblicare tutti i tipi di risorse di un archivio negozio (incluso le risorse facciata negozio, le risorse dati del negozio e i bundle di risorse) oppure solo un tipo. Per istruzioni dettagliate sulla pubblicazione delle risorse con Servizi del negozio o con la riga comandi, consultare la guida in linea di WebSphere Commerce.

Publicazione dei file di configurazione e delle risorse facciata del negozio mediante copia su WebSphere Commerce Server

Se si preferisce non impacchettare le risorse come archivio negozio, è possibile pubblicare le risorse facciata del negozio copiandole direttamente su WebSphere Commerce Server. Le risorse Web (HTML, file JSP, immagini e grafici) devono essere copiate nella root del documento dell'applicazione Web. I bundle di risorse o i file properties devono essere copiati nel percorso delle proprietà dell'applicazione.

Per copiare le risorse facciata del negozio e i file di configurazione in WebSphere Commerce Server, procedere come segue:

1. Copiare i file JSP, HTML, i file di inclusione, le immagini e i grafici nella directory del negozio (*dirnegozio*) nella root del documento di Applicazioni Web - negozi:

- **NT** unità:\WebSphere\AppServer\installedApps\
WC_Enterprise_App_ *nomeistanza*.ear\wcstores.war*dirnegozio*
- **2000** unità:\Program Files\WebSphere\AppServer\
installedApps\WC_Enterprise_App_ *nomeistanza*.ear\wcstores.war*dirnegozio*
- **AIX** /usr/WebSphere/AppServer/installedApps/
WC_Enterprise_App_ *nomenegozio*.ear/wcstores.war/*dirnegozio*
- **Solaris** /opt/WebSphere/AppServer/installedApps/
WC_Enterprise_App_ *nomeistanza*.ear/wcstores.war/*dirnegozio*
- **Linux** /opt/WebSphere/AppServer/installedApps/
WC_Enterprise_App_ *nomeistanza*.ear/wcstores.war/*dirnegozio*
- **400** /QIBM/UserData/WebASAdv4/WASinstancename/installedApps/
WC_Enterprise_App_ *nomeistanza*.ear/wcstores.war/*dirnegozio*

dove *dirnegozio* è il valore della colonna DIRECTORY della tabella di database STORE. Se questo valore non esiste, è possibile aggiungere un valore come segue: selezionare la directory del negozio=*aggiungere il nome directory relativo per le risorse file di pubblicazione*

2. Copiare i bundle di risorse e i file properties nel percorso delle proprietà dell'applicazione:

- **NT** unità:\WebSphere\AppServer\installedApps\
WC_Enterprise_App_ *nomeistanza*.ear\wcstores.war\WEB-
INF\classes*dirnegozio*
- **2000** unità:\Program Files\WebSphere\AppServer\
installedApps\WC_Enterprise_App_ *nomeistanza*.ear\wcstores.war\WEB-
INF\classes*dirnegozio*
- **AIX** /usr/WebSphere/AppServer/installedApps/
WC_Enterprise_App_ *nomeistanza*.ear/wcstores.war/WEB-
INF/classes/*dirnegozio*
- **Solaris** /opt/WebSphere/AppServer/installedApps/
WC_Enterprise_App_ *nomeistanza*.ear/wcstores.war/WEB-
INF/classes/*dirnegozio*
- **Linux** /opt/WebSphere/AppServer/installedApps/
WC_Enterprise_App_ *nomeistanza*.ear/wcstores.war/WEB-
INF/classes/*dirnegozio*

- `> 400 /QIBM/UserData/WebASAdv4/WASinstancename/installedApps/WC_Enterprise_App_nomeistanza.ear/wcstores.war/WEB-INF/classes/dirnegozio`
3. Copiare i file di configurazione nelle ubicazioni definite nel file di configurazione di WebSphere Commerce, `instance_name.xml`. Tale file risiede nella seguente directory:

- `> NT unità:\WebSphere\CommerceServer\instances\nomeistanza\xml`
- `> 2000 unità:\Program Files\WebSphere\CommerceServer\instances\nomeistanza\xml`
- `> AIX /usr/WebSphere/CommerceServer/instances/nomeistanza/xml`
- `> Solaris /opt/WebSphere/CommerceServer/instances/nomeistanza/xml`
- `> Linux /opt/WebSphere/CommerceServer/instances/nomeistanza/xml`
- `> 400 /QIBM/UserData/WebCommerce/instances/nomeistanza/xml`

I file di configurazione del negozio vengono copiati nelle seguenti ubicazioni:

- `runtime_xml.zip` viene copiato in `StoresXMLPath`. Questo percorso è definito nel file di configurazione di WebSphere Commerce, `instance_name.xml`.
 - `tools_properties.zip` viene copiato in `ToolsStoresPropertiesPath`. Questo percorso è definito nel file di configurazione di WebSphere Commerce, `instance_name.xml`.
 - `tools_xml.zip` viene copiato in `ToolsStoresXMLPath`. Questo percorso è definito nel file di configurazione di WebSphere Commerce, `instance_name.xml`.
4. Avviare il negozio mediante uno dei seguenti metodi:
- Utilizzare il comando `StoreCatalogDisplay`:
`StoreCatalogDisplay?storeId=Idnegozio&catalogId=Idcatalogo&langId=Idlingua`
dove
 - `Idnegozio` è il valore presente nella colonna `STORE_ID` della tabella di database `STORE`,
 - `Idcatalogo` è il valore presente nella colonna `CATALOG_ID` della tabella di database `CATALOG`,
 - `Idlingua` è il valore della colonna `LANGUAGE_ID` della tabella di database `LANGUAGE` per una particolare locale. Per un elenco dei valori di WebSphere Commerce predefiniti, fare riferimento alla tabella di database `LANGUAGE`.
 - Se il negozio si basa su un negozio di esempio di WebSphere Commerce, assemblare l'URL del negozio modificando il file `index.jsp` in:
 - `> NT unità:\WebSphere\CommerceServer\wc.ear\wcstores.war\dirnegozio`
 - `> 2000 unità:\Program Files\WebSphere\CommerceServer\wc.ear\wcstores.war\dirnegozio`
 - `> AIX /usr/WebSphere/CommerceServer/wc.ear/wcstores.war/dirnegozio`

- Solaris Linux /opt/WebSphere/CommerceServer/wc.ear/wcstores.war/*dirnegozio*
- 400 /QIBM/ProdData/WebCommerce/wc.ear/wcstores.war/*dirnegozio*

Aggiungere i valori corretti per i seguenti parametri:

- nomehost è il nome completo della macchina WebSphere Commerce,
- Idnegozio è il valore presente nella colonna STORE_ID della tabella di database STORE,
- Idcatalogo è il valore presente nella colonna CATALOG_ID della tabella di database CATALOG,
- Idlingua è il valore della colonna LANGUAGE_ID della tabella di database LANGUAGE per una particolare locale. Per un elenco dei valori di WebSphere Commerce predefiniti, fare riferimento alla tabella di database LANGUAGE.

Per visualizzare il negozio in un browser, avviare il seguente URL:
[http://nome_host/webapp/wcs/stores/servlet/*dirnegozio*/index.jsp](http://nome_host/webapp/wcs/stores/servlet/<i>dirnegozio</i>/index.jsp)

Parte 8. Aggiunta delle funzioni di WebSphere Commerce al negozio

Per aggiungere al negozio alcune funzioni disponibili in WebSphere Commerce, è necessario completare alcune operazioni manuali. I capitoli di questa sezione illustrano l'aggiunta delle seguenti funzioni al negozio:

- Capitolo 31, "Aggiunta dell'assistenza clienti a un negozio" a pagina 295
- Capitolo 32, "Aggiunta di spot di e-Marketing al negozio" a pagina 309

Capitolo 31. Aggiunta dell'assistenza clienti a un negozio

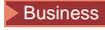
La funzione Assistenza clienti in WebSphere Commerce fornisce il supporto per l'assistenza ai clienti in tempo reale mediante un'interfaccia di testo sincrona che utilizza il server Lotus Sametime. Quando l'assistenza clienti viene abilitata nel negozio, un cliente può entrare nel negozio, fare clic su un collegamento e collegarsi a un CSR (Customer Service Representative). Quindi, un cliente può comunicare con un CSR su Internet.

Nota: Questo capitolo illustra come abilitare l'assistenza clienti nel negozio. Prima di abilitare questa funzione, è necessario installare un server Sametime e configurarlo per WebSphere Commerce. Per ulteriori informazioni, fare riferimento al manuale *WebSphere Commerce Software aggiuntivo*. E' possibile, inoltre, registrare i CSR nella Console di gestione in modo da abilitarli ad utilizzare l'assistenza clienti. Nella guida in linea di WebSphere Commerce è possibile trovare informazioni su questa attività e i concetti generali relativi all'assistenza clienti e sui CSR.

Nota:

E' possibile abilitare l'assistenza clienti nel negozio in maniera rapida e semplice mediante Servizi del negozio, se è stato creato un negozio basato su uno dei seguenti esempi:  ToolTech e NewFashion. Dopo la pubblicazione del negozio con Servizi del negozio, selezionare la visualizzazione Negozi, quindi selezionare un negozio e poi **Configura** per abilitare le funzioni di assistenza clienti. Per istruzioni più dettagliate, consultare la guida in linea di WebSphere Commerce.

Se per creare un negozio non si utilizza come base un esempio, per abilitare l'assistenza clienti è necessario eseguire alcune operazioni. La parte rimanente di questo capitolo descrive i concetti e le operazioni necessarie per abilitare l'assistenza clienti in un negozio creato in base a uno degli esempi.

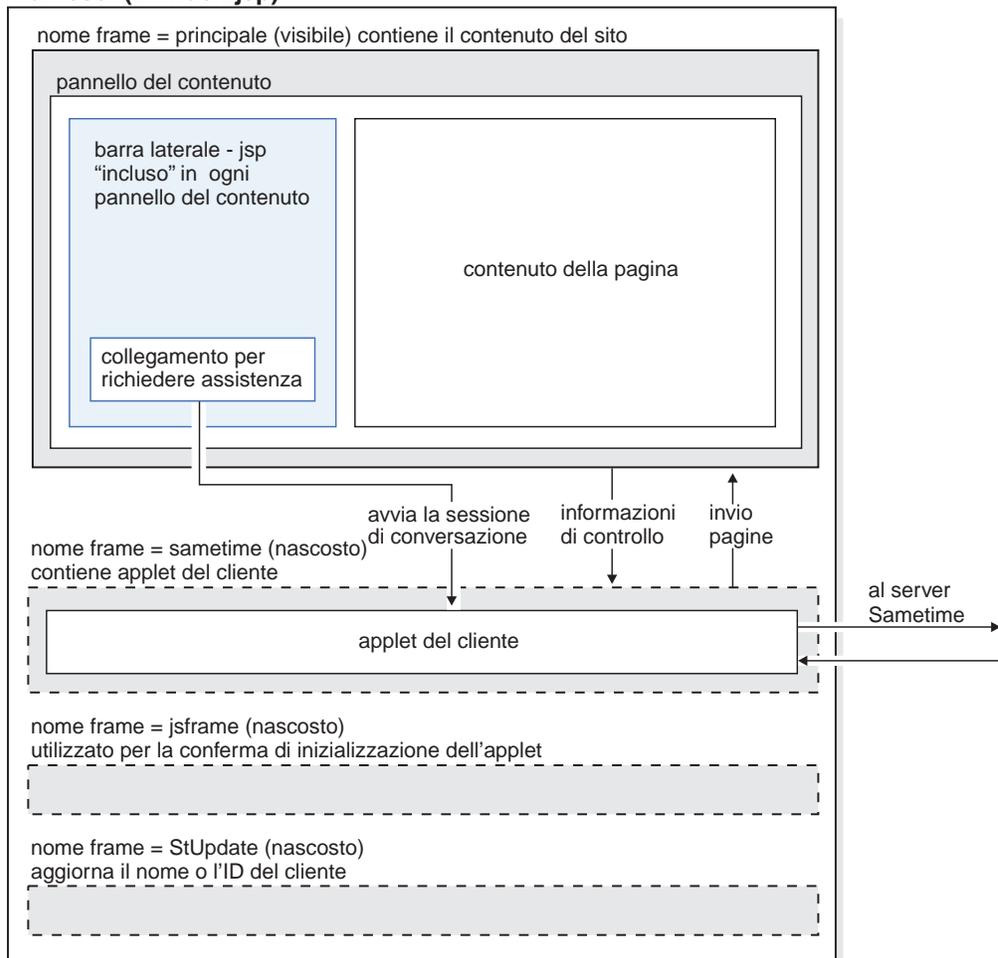
Nota: I negozi di esempio  ToolTech e NewFashion illustrano come implementare l'assistenza clienti e forniscono il codice che è possibile utilizzare nel negozio per abilitarla. Questo capitolo fa riferimento ai negozi di esempio per spiegare come abilitare l'assistenza clienti nel negozio. Assicurarsi di disporre dell'ultima versione dei negozi di esempio (disponibili sul sito Web del prodotto WebSphere Commerce).

Comprensione della funzione di assistenza clienti in un negozio

Quando un cliente seleziona il collegamento relativo all'assistenza clienti, ad esempio **Conversazione online con un responsabile dell'assistenza clienti**, in un negozio abilitato a tale funzione, viene avviata una finestra contenente un'applet. Questa applet viene eseguita all'interno di un frameset nascosto che non interferisce con l'aspetto e il comportamento del sito. Quando l'applet viene avviata, si collega al server Lotus Sametime.

Il seguente grafico illustra la composizione del frameset.

frameset (in index.jsp)



Il frameset include quattro frame:

- **Principale**: Il frame in cui è presente il contenuto del negozio, inclusi i file che creano le pagine del negozio, ovvero il corpo della pagina, i file di intestazione e piè di pagina e i file della barra laterale. Il contenuto di questo frame è visibile per i visitatori del negozio. Tener presente che il frame principale contiene i seguenti collegamenti al frame Sametime: un collegamento all'assistenza clienti e una alle informazioni di controllo. Le informazioni di controllo vengono discusse dettagliatamente in "Controllo dei clienti mediante l'assistenza clienti" a pagina 298.
- **Sametime**: Il frame che contiene l'applet dell'assistenza clienti. Questo frame non è visibile per i visitatori del negozio. Tuttavia, se un cliente fa clic sul collegamento per avviare l'applet, viene visualizzata la finestra dell'assistenza clienti. Questo frame invia le informazioni nel frame principale mediante la funzione di invio delle pagine.
- **jsframe**: Il frame che conferma che l'applet è stata caricata correttamente. Il contenuto di questo frame non viene visualizzato ai clienti.
- **StUpdate**: Il frame che aggiorna il nome o l'ID del cliente.

Uso del frameset

Quando si avvia l'applet dell'assistenza clienti in un frameset, il codice dell'applet viene separato da quello delle pagine del negozio. Come illustra il grafico sopra

riportato, le pagine del negozio si trovano nel frame principale del frameset, mentre il codice dell'applet è contenuto nel frame Sametime. La separazione del codice dell'applet dalle pagine di negozio riduce il traffico di rete perché l'applet viene scaricata una sola volta, al primo avvio del frameset. Se l'applet per l'assistenza clienti non fa parte del frameset, deve essere contenuta da ogni pagina del negozio e deve essere scaricata ogni volta che si accede a una nuova pagina.

Mediante un frameset è possibile mantenere la connessione con il server Sametime. Se l'applet fa parte di ogni pagina e non del frameset, ogni volta che un utente accede a una nuova pagina viene creata una nuova sessione di Sametime. Poiché l'applet per l'assistenza clienti si collega al server Sametime in modo anonimo, creando una nuova sessione ogni volta che un cliente accede a una nuova pagina non è possibile tenere traccia delle attività che i clienti eseguono nel negozio. Mediante un frameset, invece, è possibile mantenere la sessione Sametime originale del cliente le cui attività vengono reinviolate al server Sametime come modifica di attributi.

Problemi relativi all'uso dei frameset

Sebbene l'uso di un frameset sia il metodo più efficace per implementare l'assistenza clienti nel negozio, esso causa alcuni problemi:

- Singolo punto di entrata: i clienti possono utilizzare l'assistenza solo se sfogliano il negozio all'interno del framework. Analogamente, i CSR possono controllare i movimenti dei clienti solo attraverso il frameset. Per sfogliare il negozio tramite un frameset, i clienti devono accedere al sito da un unico punto di entrata, ad esempio la home page del negozio (nel caso dei negozi di esempio `index.jsp`). Se un cliente accede al negozio da un'altra pagina (ad esempio una pagina di catalogo), non entra nel frameset.
- Segnalibro: quando utilizzano il frameset i clienti possono porre segnalibri solo sull'URL principale del sito, non sulle singole pagine.
- Aggiornamento: quando un cliente è all'interno del frameset e fa clic sull'opzione per l'aggiornamento, viene riportato all'indirizzo del frame principale codificato nel frameset, ad esempio, `index.jsp`.
- Ridimensionamento della finestra del browser: se un cliente ridimensiona una finestra mentre si trova nel frameset, è possibile che il browser ricarichi automaticamente l'indirizzo di entrata. Se ciò accade, è possibile che il server Sametime venga terminato. In questa situazione browser diversi si comportano in modo diverso.
- Sicurezza: quando un cliente sfoglia un sito attraverso un frameset, ogni singolo frame, oltre al frameset (l'URL nella barra del percorso) mantiene il collegamento, insicuro (`http`, per impostazione predefinita sulla porta 80) o sicuro (`https`, per impostazione predefinita sulla porta 443). Se un cliente sfoglia il negozio tramite un collegamento insicuro tutti i frame del frameset si trovano in HTTP. In questo caso non si verificano problemi con SSL. Tuttavia, se il cliente accede a una pagina sicura (ad esempio la pagina di registrazione), il frame principale all'interno del frameset si trasforma in HTTPS, mentre gli altri frame restano non sicuri (`http`). In questa situazione, un cliente non è in grado di avviare l'applet di assistenza ai clienti. Il browser non autorizza l'avvio dell'applet perché quest'ultima (sicura, porta 443) sembra provenire da un server diverso dall'URL presente nella barra del percorso del browser (HTTP, porta 80). L'applet non può essere avviata finché l'intero frameset non diventa di nuovo sicuro, cioè, finché l'URL nella barra del percorso non punta a HTTPS. Per rendere di nuovo sicuro l'intero frameset, esistono diverse possibilità:
 - Reindirizzare l'intero frameset su un collegamento sicuro. Tuttavia, quando si reindirizza il frameset, tutte le conversazioni in corso vengono terminate perché l'applet viene trasferita su un nuovo collegamento Sametime.

- Reindirizzare l'intero frameset su un collegamento sicuro all'entrata nel sito. Questo metodo causa un rallentamento delle prestazioni ma garantisce una maggiore sicurezza e la riservatezza della navigazione per la sessione del cliente.

Un metodo di reindirizzamento del frameset su un collegamento sicuro consiste nell'aggiungere il codice di seguito riportato al file `index.html`:

```
- <html>
  <head>
    <META HTTP-EQUIV=Refresh CONTENT="0;URL=https://hostname/webapp/wcs/stores/
    servlet/NewFashion/index.jsp>
  </head>
</html>
</head>
```

Controllo dei clienti mediante l'assistenza clienti

L'assistenza clienti consente di controllare i clienti che comunicano con i CSR del negozio mediante le seguenti operazioni:

- Richiesta dell'ID o nome del cliente
- Determinazione della pagina sfogliata dal cliente
- Mantenimento della traccia degli articoli nel carrello degli acquisti

Per ottenere queste informazioni, il codice personalizzato viene aggiunto alle pagine del negozio. Le sezioni seguenti illustrano come vengono implementate queste funzioni di controllo nei negozi di esempio.

Richiesta dell'ID o nome del cliente

Una volta che l'applet di assistenza clienti viene avviata e che il CSR è collegato, quest'ultimo è in grado di identificare l'utente che utilizza l'applet dal nome o dall'ID. I negozi di esempio includono il codice specializzato che funziona con l'applet di assistenza clienti per stabilire il nome e l'ID del cliente. Questo codice determina se l'utente in questione è un cliente ospite, un cliente ospite con articoli in un carrello degli acquisti o un cliente registrato, quindi assegna un nome o un ID al cliente e lo trasmette all'applet di assistenza clienti. Questi nomi vengono poi visualizzati al CSR. Ad esempio, a un cliente ospite che non ha inserito articoli nel carrello degli acquisti viene assegnato l'ID acquirente -1002. Nel caso di un cliente ospite che ha inserito articoli nel carrello degli acquisti, l'ID acquirente viene visualizzato e se il cliente è registrato viene visualizzato nome e cognome.

I negozi di esempio ottengono il nome o l'ID del cliente aggiungendo il seguente codice al file di intestazione del negozio che aggiorna il frame `StUpdate`. Questo codice è incluso in `header.jsp` per `NewFashion` e in `NavHeader.jsp` per

 ToolTech.

Nota: Ogni volta che un cliente sfoglia una nuova pagina del negozio, viene aggiornato l'ID o il nome.

```
<script language="javascript">
  if (typeof top.updateStInfo == 'function')
    top.updateStInfo();
</script>
```

Il codice precedentemente riportato aggiorna le seguenti stringhe nel frame `StUpdate`.

```
//set Customer Name for LiveHelp if user is registered.

if (userRegistrationDataBean.findUser()) {
  if (userRegistrationDataBean.getLastName() != null & &
```

```

userRegistrationDataBean.getLastName().length() > 0) {
    if(cmdcontext != null) {
        Long uid = cmdcontext.getUserId();
        String customerName = "";
        if (locale.toString().equals("ja_JP") || locale.toString().equals("ko_KR")
|| locale.toString().equals("zh_CN") || locale.toString().equals("zh_TW"))
        {
            customerName = "" + userRegistrationDataBean.getLastName() + " "
+ userRegistrationDataBean.getFirstName();
        }
        else {
            customerName = "" + userRegistrationDataBean.getFirstName() + " "
+ userRegistrationDataBean.getLastName();
        }
    }
    else {
        customerName=userRegistrationDataBean.getUserId();
        if (customerName.equals("-1002"))
            customerName="";
        customer_name=customer_name.trim();
    }
}

```

Nella pagina di scollegamento del negozio di esempio, è incluso un ulteriore codice che imposta il nome del cliente su un ID generato e reimposta il numero di articoli nel carrello degli acquisti su zero. La pagina di scollegamento in NewFashion è LoginForm.jsp. In  ToolTech è Logoff.jsp. Il codice personalizzato è il seguente:

```

<HTML>
<HEAD>
<SCRIPT language="javascript">
    if (typeof parent.setCustomerName == 'function')
        parent.setCustomerName (parent.WCSGUESTID, '')
    if (typeof parent.setShoppingCartItems == 'function')
        parent.setShoppingCartItems(0);
</SCRIPT>
</HEAD>
</HTML>

```

Determinazione della pagina sfogliata dal cliente

L'assistenza clienti consente ai CSR di determinare quale pagina del negozio sta sfogliando un cliente in un determinato momento. I negozi di esempio determinano quali pagine stanno visualizzando i clienti aggiungendo il seguente codice al file di intestazione (header.jsp in NewFashion e in NavHeader.jsp in

 ToolTech):

```

<%
//Determine Page Type for LiveHelp
String headerType = (String) request.getAttribute("liveHelpPageType");
if (headerType==null)
    headerType = "";
%>

<script language="javascript">
<%
String pname = request.getRequestURI();
int indpn = pname.lastIndexOf('/');
indpn = pname.lastIndexOf('/', indpn-1);
if(indpn != -1)
    pname = pname.substring(indpn+1);

//Determine if this is a personal page or not
if (headerType.equals("personal") ) {
%>

```

```

    if (typeof parent.setPageParams == 'function')
        parent.setPageParams('PERSONAL_URL', '<%=pname%>');
    <% } else { %>
    if (typeof parent.setPageParams == 'function')
        parent.setPageParams(location.href, '<%=pname%>');
    <% } %>
</script>

```

Nelle pagine che non utilizzano il file di intestazione viene incluso il file StHeader1.jsp. StHeader1.jsp contiene lo stesso codice aggiunto al file di intestazione del negozio.

Per preservare la riservatezza dei clienti, i CSR non devono avere accesso ad alcune pagine. Ad esempio, un CSR non può avere accesso a una pagina di campagna, che riporta un prezzo stabilito da un contratto, o alle pagine che contengono l'ID utente, ad esempio la pagina della rubrica. Queste pagine sono contrassegnate come personali. Nei negozi di esempio le seguenti pagine sono contrassegnate come personali:

- NewFashion
 - AddressBookForm.jsp
 - AllocationCheck.jsp
 - edit_registration.jsp
 - emptyshoppingcart.jsp
 - interestItemDisplay.jsp
 - myaccount.jsp
 - orderItemDisplay.jsp
 - OrderDisplayPending.jsp
 - ResultList.jsp
 - shoppingcart.jsp
 - TrackOrderStatus.jsp
- Business ToolTech
 - Address.jsp
 - Addressbook.jsp
 - AddToExistReqList.jsp
 - AdvancedSearch.jsp
 - AllocationCheck.jsp
 - CatalogMainDisplay.jsp
 - CatalogItemDisplay.jsp
 - CatalogTopCategoriesDisplay.jsp
 - Confirmation.jsp
 - OrderDisplayPending.jsp
 - OrderItemDisplay.jsp
 - OrderDetail.jsp
 - QuickOrder.jsp
 - RequisitionListCreate.jsp
 - RequisitionListDetailDisplay.jsp
 - RequisitionListDisplay.jsp
 - RequisitionListUpdate.jsp
 - Result List.jsp

- Shipping.jsp
- shoppingcart.jsp
- TrackOrderStatus.jsp
- UserAccount.jsp
- UserRegistrationUpdate.jsp

Per contrassegnare una pagina come personale, ovvero non disponibile per il CSR, i negozi di esempio includono nella pagina il seguente codice, prima dell'intestazione.

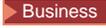
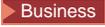
```
<%
// Set header type needed for this JSP for Customer Care. This must
// be set before Header.jsp
request.setAttribute("liveHelpPageType", "personal");
%>

<%
String incfile;
incfile = includeDir + "Header.jsp";
%>
<jsp:include="<%=incfile%>" flush="true"/>
```

Nota: Sebbene un CSR non possa visualizzare una pagina contrassegnata come personale, può visualizzarne l'URL.

Mantenimento della traccia del numero degli articoli nel carrello degli acquisti

La funzione di assistenza clienti consente ai CSR di tenere traccia del numero di articoli presenti nel carrello degli acquisti di un cliente in un determinato momento. I negozi di esempio ottengono il numero di articoli nel carrello degli acquisti nelle seguenti ubicazioni:

- Pagina del carrello degli acquisti (NewFashion: shoppingcart.jsp,  ToolTech:ShoppingCart.jsp)
- Pagina del carrello degli acquisti vuoto (NewFashion: emptyshopcart.jsp,  ToolTech:EmptyOrder.jsp)
- Pagina di conferma ordini (NewFashion: confirmation.jsp,  ToolTech:confirmation.jsp)
- Pagina di scollegamento ( ToolTech:Logoff.jsp)

Nota: Sebbene il carrello degli acquisti sia stato progettato come pagina personale, i CRS possono tenere traccia del numero degli articoli nel carrello degli acquisti. Non possono visualizzare il contenuto del carrello, solo il numero di articoli. Tuttavia, un CSR può visualizzare il contenuto di un carrello degli acquisti utilizzando il pulsante **Visualizza carrello degli acquisti**. Per ulteriori informazioni, fare riferimento alla guida in linea di WebSphere Commerce.

I negozi di esempio determinano il numero di articoli nel carrello degli acquisti aggiungendo il seguente codice alle pagine sopra riportate:

- Innanzitutto una variabile int è definita


```
int liveHelpShoppingCartItems= 0;
```
- Poi, la seguente riga del codice è utilizzata per aggiungere la quantità a liveHelpShoppingCartItems in qualsiasi momento si verifichi l'aggiunta di un articolo d'ordine al carrello:


```
liveHelpShoppingCartItems+= orderItem.getQuantityInEJBType().intValue();
```

- A questo punto, il seguente codice viene aggiunto alla fine della pagina per impostare il nome del cliente sull'ID dell'acquirente ospite e ottenere il numero di articoli presenti nel carrello degli acquisti del cliente.

```
<script language="javascript">
  if (typeof parent.setShoppingCartItems == 'function')
    parent.setShoppingCartItems(<%=liveHelpShoppingCartItems%>);
</script>
```

Il seguente codice viene utilizzato nella pagina del carrello degli acquisti vuoto e nella pagina di conferma ordine per reimpostare il numero di articoli presenti nel carrello su zero:

```
<script language="javascript">
  if (typeof parent.setShoppingCartItems == 'function')
    parent.setShoppingCartItems(0);
</script>
```

Aggiunta dell'assistenza clienti a un negozio

Per aggiungere l'assistenza clienti a un negozio non basato su un esempio, procedere come segue:

Parte 1: Prerequisiti per l'installazione

Affinché l'assistenza clienti possa funzionare nel negozio, procedere come segue:

- Installare un server Sametime. Per ulteriori informazioni, consultare il manuale *WebSphere Commerce Software aggiuntivo*.
- Installare il pacchetto di integrazione di WebSphere Commerce Sametime. Per ulteriori informazioni, consultare il manuale *WebSphere Commerce Software aggiuntivo*.
- Arrestare l'istanza di WebSphere Commerce, abilitare Sametime nel Gestore configurazione, quindi riavviare l'istanza. Per ulteriori informazioni, consultare il manuale *WebSphere Commerce Software aggiuntivo*.
- Creare un CSR e registrarlo per l'assistenza clienti mediante la console di gestione. Per ulteriori informazioni, fare riferimento alla guida in linea di WebSphere Commerce.

Parte 2: Copia dei file di integrazione dell'assistenza clienti dal negozio di esempio

I negozi di esempio NewFashion e ToolTech includono i seguenti file che vengono utilizzati per integrare l'assistenza clienti nel negozio di esempio:

- *Sametime.js*: contiene le funzioni JavaScript incluse per tutti i frame. I nomi di queste funzioni corrispondono alle pagina del frame con il suffisso *parent*, ad esempio *parent.setCustomerName*.
- *StBlank.jsp*: un file JSP vuoto.
- *StFrame.jsp*: contiene le funzioni JavaScript e l'applet dell'assistenza clienti per la facciata del negozio.
- *StReadyJS.jsp*: indica che l'applet è stata caricata correttamente.
- *StHeader1.jsp*: un file di intestazione trasferito all'applet in un parametro e indicante se la pagina che include l'intestazione è personale.
- *StUpdate.jsp*: aggiorna l'ID e le informazioni sul nome del cliente.

Per copiare i file di integrazione di Sametime dal negozio di esempio al negozio personalizzato, procedere come segue:

1. Individuare il file di archivio negozio per NewFashion o ToolTech. I file degli archivi negozio si trovano nelle seguenti directory:
 -  unità:\WebSphere\CommerceServer\samplstores
 -  unità:\Program Files\WebSphere\CommerceServer\samplstores
 -  /usr/WebSphere/CommerceServer/samplstores
 -  /opt/WebSphere/CommerceServer/samplstores
 -  /opt/WebSphere/CommerceServer/samplstores
 -  /qibm/proddata/WebCommerce/samplstores
2. Aprire la cartella di ToolTech o NewFashion, quindi selezionare un archivio negozio ToolTech o NewFashion.
3. Aprire il file dell'archivio negozio utilizzando WinZip o uno programma simile.
4. Individuare il file webapp.zip. Aprirlo con WinZip o un programma simile.
5. Selezionare i seguenti file:
 - Sametime.js
 - StBlank.jsp
 - StFrame.jsp
 - StReadyJS.jsp
 - StHeader1.jsp
 - StUpdate.jsp
6. Estrarre i file nella directory che contiene le risorse web per il negozio.

Nota: StHeader1.jsp è una directory di inclusione negli archivi di esempio. Se esiste una directory separata per i file inclusi nelle pagine del negozio, salvare StHeader1.jsp in tale directory. Altrimenti, salvare il file nella directory in cui risiedono gli altri file del negozio.

Parte 3: Aggiunta del frameset al negozio

Come discusso in "Uso del frameset" a pagina 296 l'applet di assistenza clienti viene eseguita in un frameset. Questo frameset deve essere aggiunto alla home page del negozio o alla pagina dalla quale è più probabile che i clienti accederanno al negozio. Per aggiungere il frameset al negozio, procedere come segue:

1. Stabilire quale pagina è il punto di entrata del negozio.
2. Aprire il file dell'archivio negozio per ToolTech o NewFashion. I file degli archivi negozio si trovano nelle seguenti directory:
 -  unità:\WebSphere\CommerceServer\samplstores
 -  unità:\Program Files\WebSphere\CommerceServer\samplstores
 -  /usr/WebSphere/CommerceServer/samplstores
 -  /opt/WebSphere/CommerceServer/samplstores
 -  /opt/WebSphere/CommerceServer/samplstores
 -  /qibm/proddata/WebCommerce/samplstores
3. Aprire il file dell'archivio negozio utilizzando WinZip o uno programma simile.
4. Individuare il file webapp.zip. Aprirlo con WinZip o un programma simile.
5. Aprire il file index.jsp.
6. Copiare il seguente codice:

```

<script src="<%=Sametime.js%"></script>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"DTD/xhtml1-transitional.dtd">
<html>
<head>
<title></title>
<script language="javascript">
var MainPageURL="";
if (MainPageURL=="")
MainPageURL="/webapp/wcs/stores/servlet/Logoff?storeId=<%=storeId%>
&langId=<%=langId%>&URL=LogonForm?
storeId=<%=storeId%>&catalogId=<%=catalogId%>";
function loadFrame()
{
main.document.location.href=MainPageURL;
}
</script>
</head>
<%
String sBasePath="/webapp/wcs/stores/servlet/";
String sSametimeUrl="StFrame.jsp?storeId="+storeId;
String sBlankUrl="StBlank.jsp?storeId="+storeId;
String sUpdateUrl="StUpdate.jsp?storeId="+storeId;
%>
<FRAMESET border=0 frameBorder=0 ROWS="100%,1,1,1,1,1" onLoad="loadFrame();">
<FRAME NAME="main"
SRC="javascript:top.loadFrame();" MARGINWIDTH=0 SCROLLING="Auto"
FRAMEBORDER="no" noresize>
<FRAME NAME="JSFrame"
SRC="<%=sBlankUrl%>" MARGINWIDTH=0 SCROLLING="no"
FRAMEBORDER="no" noresize>
<FRAME NAME="sametime" SRC="<%=sSametimeUrl%>" MARGINWIDTH=0 SCROLLING="no"
FRAMEBORDER="no" noresize>
<FRAME NAME="StUpdate" SRC="<%=sUpdateUrl%>" MARGINWIDTH=0 SCROLLING="no"
FRAMEBORDER="no" noresize>
</FRAMESET>
</html>

```

Nota: Nell'esempio sopra riportato, ricavato dal negozio di esempio ToolTech, l'origine (SRC) del frame "principale" è un comando per un'altra pagina. A seconda dell'impostazione del negozio, l'SRC può essere un comando, un file JSP o un file HTML.

7. Incollare il codice copiato al passo 6 nella pagina designata come punto di entrata del negozio. Apportare le necessarie modifiche all'origine (SRC) per il frame principale.
8. Salvare il file.

Parte 4: Aggiunta del codice per ottenere l'ID o il nome del cliente

Per visualizzare il nome dell'ID acquirente del cliente che utilizza l'assistenza clienti nel CSR, procedere come segue:

1. Fare riferimento alle informazioni riportate in "Richiesta dell'ID o nome del cliente" a pagina 298.
2. Decidere da dove ricevere il nome del cliente o l'ID acquirente. Ad esempio:
 - Punto di entrata nel negozio
 - Registrazione nuova o aggiornata
 - Scollegamento
 - Intestazione

3. Dopo aver stabilito da dove ricevere il nome del cliente o l'ID acquirente, determinare se tali pagine includono il file di intestazione principale del negozio. Se non lo includono, stabilire se aggiungere il codice necessario alla pagina stessa o includere un file di intestazione.
4. Dal file webapp.zip nell'archivio negozio NewFashion o ToolTech, aprire i seguenti file:
 - NewFashion: header.jsp
 - ToolTech: NavHeader.jsp

Nota: Entrambi i file risiedono nella directory di inclusione nel file webapp.zip.

5. Copiare il seguente codice:

```
<script language="javascript">
  if (typeof top.updateStInfo == 'function')
    top.updateStInfo()
</script>
```

6. Incollare il codice copiato al passo 5 nel file di intestazione del negozio o direttamente nelle pagine appropriate.

Nota: Se si preferisce, è possibile includere il file StHeader1.jsp della pagina appropriata invece di copiare il codice sopra riportato nel file di intestazione del negozio o direttamente nel file.

7. Salvare i file.
8. (Facoltativo) Per reimpostare l'ID cliente nella pagina di scollegamento (ad esempio dall'ID del cliente a -1002) e per reimpostare il numero di articoli nel carrello degli acquisti su zero, aggiungere il seguente codice al file di scollegamento:

```
<HTML>
<HEAD>
<SCRIPT language="javascript">
  if (typeof parent.setCustomerName == 'function')
    parent.setCustomerName (parent.WCSGUESTID, '')
  if (typeof parent.setShoppingCartItems == 'function')
    parent.setShoppingCartItems(0);
</SCRIPT>
</HEAD>
</HTML>
```

Parte 5: Aggiunta del codice per determinare quale pagina sta sfogliando il cliente

Per determinare quale pagina sta sfogliando il cliente, procedere come segue:

1. Includere il file StHeader1.jsp al file di intestazione del negozio, ad esempio:


```
<%@ include file="StHeader1.jsp" %>
```
2. Aggiungere il seguente codice alle pagine che devono essere contrassegnate come personali e non essere pertanto disponibili per il CSR:

```
<%
// Set header type needed for this JSP for Customer Care. This must
// be set before Header.jsp
request.setAttribute("liveHelpPageType", "personal");
%>

<%
String incfile;
incfile = includeDir + "Header.jsp";
%>
<jsp:include="<%=incfile%>" flush="true"/>
```

3. Aggiungere il seguente codice alle pagine che non utilizzano un'intestazione ma devono essere contrassegnate come personali:

```
<%
// Set header type needed for this JSP for Customer Care. This must
// be set before StHeader1.jsp
request.setAttribute("liveHelpPageType", "personal");

String incfile;
incfile = includeDir + "StHeader1.jsp";
%>
<jsp:include page="<%=incfile%>" flush="true"/>
```

Parte 6: Aggiunta del codice per tenere traccia del numero di articoli nel carrello degli acquisti

Per consentire ai CSR di tenere traccia degli articoli presenti nel carrello degli acquisti di un cliente, procedere come segue:

1. Fare riferimento alle informazioni riportate in "Mantenimento della traccia del numero degli articoli nel carrello degli acquisti" a pagina 301.
2. Stabilire dove tenere traccia degli articoli del carrello degli acquisti. Ad esempio:
 - Pagina Carrello degli acquisti
 - Pagina del carrello degli acquisti vuoto
 - Pagina Conferma ordini
 - Pagina di scollegamento
3. Nelle pagine in cui si desidera tenere traccia degli articoli del carrello degli acquisti, procedere come segue:

- a. Definire una variabile int:

```
int liveHelpShoppingCartItems= 0;
```

- b. Per aggiungere la quantità a shoppingCartItems quando un articolo d'ordine viene aggiunto al carrello, aggiungere la seguente riga del codice:

```
liveHelpShoppingCartItems+= orderItem.getQuantityInEJBType().intValue();
```

- c. Alla fine della pagina, aggiungere il seguente codice, per impostare l'ID acquirente ospite (-1002) sull'ID acquirente effettivo e ottenere il numero di articoli nel carrello degli acquisti:

```
<script language="javascript">
if (typeof parent.setCustomerName == 'function')
    parent.setCustomerName(<%=cmdcontext.getUserId()%>, parent.CustomerName);
if (typeof parent.setShoppingCartItems == 'function')
    parent.setShoppingCartItems(<%=liveHelpShoppingCartItems%>);
</script>
```

- d. Se si desidera tenere traccia dei dati nella pagina del carrello degli acquisti vuoto e nella pagina di conferma ordini, aggiungere il seguente codice a tali pagine per reimpostare il valore del carrello degli acquisti su zero:

```
<script language="javascript">
if (typeof parent.setShoppingCartItems == 'function')
    parent.setShoppingCartItems(0);
</script>
```

Parte 7: Aggiunta di un collegamento all'assistenza clienti

Per consentire ai clienti di accedere all'assistenza clienti del negozio, procedere come segue:

1. Stabilire dove inserire il collegamento all'assistenza clienti. Ad esempio, è possibile inserire il collegamento in una barra di navigazione affinché sia sempre disponibile ai clienti o solo in alcune pagine del negozio.
2. Copiare il seguente codice nelle pagine che conterranno il collegamento:

```
<a href="javascript:if((parent.sametime != null)) top.interact();">  
<%=infashiontext.getString("LiveHelp")%></a>
```

Parte 8: Modifica dei messaggi che vengono visualizzati al cliente

I messaggi che vengono visualizzati a un cliente quando si collega a un CSR, ad esempio, "Salve, in cosa posso esserle utile?" oppure "L'ufficio è aperto dalla 9.00 alle 21.00" sono memorizzati nei file `properties` sul server Sametime. I file `properties` sono suddivisi in due tipi: `Customer.properties` e `Agent.properties`. Il file `Customer.properties` contiene i messaggi che vengono visualizzati al cliente, mentre il file `Agent.properties` contiene le informazioni visualizzate al CSR. Per entrambi esistono file specifici della locale corrispondenti, ad esempio `Customer_de_DE.properties` e `Agent_de_DE.properties`, per ciascuna locale installata nell'istanza di WebSphere Commerce.

Per modificare i messaggi contenuti in questi file, procedere come segue:

1. Individuare i file `properties` sul server Sametime. Per impostazione predefinita, questi file si trovano nella seguente directory:
 -  unità:\Sametime\Data\domino\html\wc\properties
2. Apportare le necessarie modifiche.
3. Chiudere e salvare il file.

Capitolo 32. Aggiunta di spot di e-Marketing al negozio

Gli spot di e-Marketing si riservano dello spazio sulle pagine del negozio in cui viene visualizzato il contenuto di marketing personalizzato per le iniziative di campagna. Quando viene richiesta una pagina da un cliente, gli spot di e-Marketing visualizzati nella pagina comunicheranno con il server di regole per elaborare il codice associato allo spot. Ogni spot di e-Marketing viene associato ad una o più iniziative di campagna. Per ulteriori informazioni su campagne e iniziative, consultare Capitolo 12, "Risorse Campagne" a pagina 117 e la guida in linea di WebSphere Commerce.

Per visualizzare correttamente le iniziative delle campagne nelle pagine di negozio, è necessario aggiungere uno spot di e-Marketing al file JSP e quindi registrarlo nel database mediante WebSphere Commerce Accelerator. Questo capitolo discute come aggiungere spot di e-Marketing ai file JSP del negozio. Per ulteriori informazioni sulla registrazione dello spot di e-Marketing nel database mediante WebSphere Commerce Accelerator, fare riferimento alla guida in linea di WebSphere Commerce.

Spot di e-Marketing

Il seguente è un esempio di spot di e-Marketing:

```
<!-- =====
/**-----
/** The sample contained herein is provided to you "AS IS".
/**
/** It is furnished by IBM as a simple example and has not been thoroughly tested
/** under all conditions. IBM, therefore, cannot guarantee its reliability,
/** serviceability or functionality.
/**
/** This sample may include the names of individuals, companies, brands and
/** products
/** in order to illustrate concepts as completely as possible.
/**All of these names
/** are fictitious and any similarity to the names and addresses used by actual
    persons
/**or business enterprises is entirely coincidental.
/**-----
/**
=====-->
<%
/**
 * START - the following code should exist only once in a page, it initialize the
 *         command context and store data bean.
 */

// create the store bean to get the store directory
String collateralPath = "/webapp/wcs/stores/";
com.ibm.commerce.command.CommandContext emsCommandContext =
(com.ibm.commerce.command.CommandContext) request.getAttribute(
ECCConstants.EC_COMMANDCONTEXT);
com.ibm.commerce.common.beans.StoreDataBean storeDataBean =
new com.ibm.commerce.common.beans.StoreDataBean();
storeDataBean.setStoreId(emsCommandContext.getStoreId().toString());
com.ibm.commerce.beans.DataBeanManager.activate(storeDataBean, request);
if (storeDataBean.getDirectory() != null) {
    collateralPath += storeDataBean.getDirectory() + "/";
}
%>
<!-- =====
// The following HTML form submits the request on the e-marketing spot to the
```

```

ClickInfo
// command which captures the campaign statistics, and redirect to the location
//specified by the URL parameter.
===== -->
<form name="storeEmsForm" method="POST" action="/webapp/wcs/stores/servlet/ClickInfo">
  <input type="hidden" name="evtype">
  <input type="hidden" name="mpe_id">
  <input type="hidden" name="intv_id">
  <input type="hidden" name="URL">
</form>
<%
/**
 * END - the following code should exist only once in a page, it initialize the
 *       command context and store data bean.
 */
%>
<%
/**
 * START - the following code can be used to drop multiple e-marketing
spots onto the page.
 *       Customize the appropriate EMarketingSpot instance name and the
e-marketing spot
 *       name before use. Duplicate this code if more than 1 spot is
needed, do not use
 *       the same spot name.
 */

// create the e-Marketing Spot
com.ibm.commerce.marketing.beans.EMarketingSpot eMarketingSpot =
new com.ibm.commerce.marketing.beans.EMarketingSpot();

// IMPORTANT - set the correct name here
eMarketingSpot.setName("eMarketingSpotName");

// the maximum number of products/categories/ad copies that display
//through this e-marketing spot can be set here
eMarketingSpot.setMaximumNumberOfCatalogEntries(20);
eMarketingSpot.setMaximumNumberOfCategories(20);
eMarketingSpot.setMaximumNumberOfCollateral(20);

// instantiate the bean
com.ibm.commerce.beans.DataBeanManager.activate(eMarketingSpot,
request);
%>

<%
// The following block is used to display the advertisements associated with this
// e-marketing spot. The URL link defined with an advertisement can be referenced
//through the submission of the HTML form attached above.

if (eMarketingSpot.getCollateral() != null && eMarketingSpot.getCollateral().
length > 0) {
%>
<TABLE>
<% for (int i=0; i eMarketingSpot.getCollateral().length; i++) { %>
<TR>
<% if (eMarketingSpot.getCollateral()[i].getTypeName().equals("Image")) { %>
  <TD>
    A HREF="javascript:document.storeEmsForm.evtype.value='CpgnClick'
;document.storeEmsForm.mpe_id.value=' <%= eMarketingSpot.getId() %>
';document.storeEmsForm.intv_id.value='<%= eMarketingSpot.getCollateral
()[i].getInitiativeId() %>';document.storeEmsForm.URL.value='
<%= eMarketingSpot.getCollateral()[i].getUrlLink() %>';
document.storeEmsForm.submit();"
    IMG SRC=" <%= collateralPath + eMarketingSpot.getCollateral()[i].getLocation() %>">
  </A>
</TD>
  <TD>
    <%= eMarketingSpot.getCollateral()[i].getMarketingText() %>
  </TD>
<% } else if (eMarketingSpot.getCollateral()[i].getTypeName().equals("Flash")) { %>
  <TD>

```

```

        EMBED src=" <%= collateralPath + eMarketingSpot.getCollateral()[i].getLocation()
%>" quality=high bgcolor=#FFFFFF WIDTH=120 HEIGHT=90
TYPE="application/x-shockwave-flash"> </EMBED>
    </TD>
    <TD>
        A HREF="javascript:document.storeEmsForm.evtype.value='CpgnClick'
;document.storeEmsForm.mpe_id.value='<%= eMarketingSpot.getId()
%>';document.storeEmsForm.intv_id.value='<%= eMarketingSpot.getCollateral()
[i].getInitiativeId() %>';document.storeEmsForm.URL.value='
<%= eMarketingSpot.getCollateral()[i].getUrlLink() %>';
document.storeEmsForm.submit();">
        <%= eMarketingSpot.getCollateral()[i].getMarketingText() %>
    </A>
    </TD>
<%= } %>
</TR>
<%= } %>
</TABLE>
<%= } %>

<%=
// The following block is used to display the categories associated with this e-marketing
// spot. The category display page which shows the selected category in the campaign will
// be referenced through the submission of the HTML form attached above.

if (eMarketingSpot.getCategories() != null && eMarketingSpot.getCategories().length > 0) {
%>
<TABLE>
<%= for (int i=0; i eMarketingSpot.getCategories().length; i++) { %>
<TR>
<TD>
        A HREF="/webapp/wcs/stores/servlet/ClickInfo?evtype=CpgnClick
&mpe_id=<%= eMarketingSpot.getId() %>&intv_id=<%= eMarketingSpot.
getCategories()[i].getInitiativeId()
%>%URL=/webapp/wcs/stores/servlet/CategoryDisplay&
<%= ECConstants.EC_STORE_ID %>=<%= eMarketingSpot.getStoreId().toString() %>&
<%= ECConstants.EC_CATEGORY_ID %>=<%=
eMarketingSpot.getCategories()[i].getCategoryId() %>&<%=
= ECConstants.EC_CATALOG_ID %>=<%= eMarketingSpot.getCategories()[i].getCatalogId() %>
&<%= ECConstants.EC_LANGUAGE_ID %>=<%= eMarketingSpot.getCategories()[i].getDescription
(emsCommandContext.getLanguageId()).getName() %>
</A>
</TD>
<TD><%= eMarketingSpot.getCategories()[i].getDescription
(emsCommandContext.getLanguageId()).getLongDescription() %> </TD>
</TR>
<%= } %>
</TABLE>
<%= } %>

<%=
// The following block is used to display the products associated with this e-marketing
// spot. The product display page which shows the selected product in the campaign will
// be referenced through the submission of the HTML form attached above.

if (eMarketingSpot.getCatalogEntries() != null &&
eMarketingSpot.getCatalogEntries().length > 0) {
%>
<TABLE>
<%= for (int i=0; i eMarketingSpot.getCatalogEntries().length; i++) { %>
<TR>
<TD>
        A HREF="/webapp/wcs/stores/servlet/ClickInfo?evtype=CpgnClick
&mpe_id=<%= eMarketingSpot.getId() %>&intv_id=<%=
eMarketingSpot.getCatalogEntries()[i].getInitiativeId()
%>%URL=/webapp/wcs/stores/servlet/ProductDisplay
&<%= ECConstants.EC_STORE_ID %>=<%= eMarketingSpot.getStoreId().toString()
%>&<%= ECConstants.EC_PRODUCT_ID %>=<%=
eMarketingSpot.getCatalogEntries()[i].getCatalogEntryID() %>&<%=
= ECConstants.EC_LANGUAGE_ID %>=<%= eMarketingSpot.getCatalogEntries()[i].getCatalogEntryID() %>
%>">
        IMG SRC="<%= collateralPath + eMarketingSpot.getCatalogEntries()

```

```

[i].getDescription(emsCommandContext.getLanguageId()).
getThumbnail() %>" ALT="<%= eMarketingSpot.getCatalogEntries()[i].getDescription
(emsCommandContext.getLanguageId()).getShortDescription()%>" BORDER=0 WIDTH=60>
</A>
</TD>
<TD><%= eMarketingSpot.getCatalogEntries()[i].getDescription
(emsCommandContext.getLanguageId()).getShortDescription() %> </TD>
</TR>
<% } %>
</TABLE>
<% } %>

<% /**
 * END - the following code is used to drop multiple e-marketing spots onto the page.
 * Customize the appropriate e-marketing spot name before use.
 * Duplicate this code if more than 1 spot is needed, do not use the same spot name.
 */
%>

```

Il precedente spot di e-Marketing supporta tre tipi di iniziative di campagna:

- Consigli sul prodotto
- Consigli sulla categoria
- Campagne pubblicitarie

Nota: Per ulteriori informazioni su ciascuna di queste iniziative, consultare Capitolo 12, “Risorse Campagne” a pagina 117.

Bean e-MarketingSpot

Gli spot di e-Marketing utilizzano il bean per restituire i risultati delle iniziative di campagne pianificate sullo spot. Utilizzando proprietà diverse del bean è possibile personalizzare lo spot di e-Marketing e l’iniziativa di campagna corrispondente. Per ulteriori informazioni sul bean e-MarketingSpot e le relative proprietà, fare riferimento alla guida in linea di WebSphere Commerce.

Aggiunta di uno spot di e-Marketing alle pagine del negozio

Per aggiungere uno spot di e-Marketing alle pagine del negozio, procedere come segue:

1. Stabilire su quali file JSP sarà visualizzato lo spot. È possibile aggiungerlo a più file JSP.
2. Stabilire in quale punto del file JSP inserire lo spot.
3. Copiare lo spot di e-Marketing di esempio in “Spot di e-Marketing” a pagina 309.
4. Incollare lo spot di e-Marketing di esempio nella posizione desiderata del file JSP.
5. Personalizzare lo spot di e-Marketing di esempio in modo da adattarlo al layout del file JSP.
6. Nel codice dello spot di e-Marketing, assegnare un nome allo spot.

Nota: Gli spot di e-Marketing devono essere in maniera descrittiva, in modo da includere la posizione, come ad esempio HomePageAd o CheckoutPageRecommendation. Ciò consente di individuare più facilmente la posizione in cui verranno visualizzati ed il tipo di contenuto associato. Inoltre, per differenziare due spot di e-Marketing che vengono visualizzati nella stessa pagina, è possibile aggiungere numeri al nome. I nomi degli spot di e-Marketing devono essere degli

identificativi validi per Java. Quando si registra lo spot di e-Marketing nel database con WebSphere Commerce Accelerator, è necessario utilizzare questo nome.

7. Se si desidera più di uno spot di e-Marketing per file JSP, ripetere le operazioni da 2 a 6. Quando si aggiunge un altro spot di e-Marketing a un file JSP, assicurarsi di copiare tutta la seconda sezione dello spot di e-Marketing di esempio fornito in "Spot di e-Marketing" a pagina 309 e di incollarla di nuovo sul file JSP. Quindi, modificare il nome dello spot di e-Marketing nel file JSP.
8. Registrare lo spot di e-Marketing nel database mediante WebSphere Commerce Accelerator. Per istruzioni dettagliate, consultare la guida in linea di WebSphere Commerce.

Nota:

- a. Se si intende aggiungere l'ID negozio, l'ID catalogo o l'ID lingua all'URL mediante la seguente convenzione, "langId=<%= languageId %>", tener presente che il file in cui viene incorporato lo spot di e-marketing deve rendere disponibile l'ID appropriato. Gli ID possono essere richiamati anche tramite il contesto di comando, ad esempio `getCommandContext().getLanguageId(?)`.
- b. A causa della struttura del negozio di esempio NewFashion, solo i prodotti possono essere consigliati mediante lo spot di e-Marketing, non gli articoli dei negozi che si basano sul negozio di esempio NewFashion.
- c. Il parametro URL, `CatalogDisplay` deve cominciare con "&" invece che con "?" perché il codice non fa riferimento direttamente al comando.

Parte 9. Appendici

Appendice A. Legenda UML

Unified Modeling Language è un linguaggio grafico standard per la presentazione di diversi elementi di progettazione software. Gli esempi che seguono rappresentano alcuni degli elementi più comuni di UML. Per ulteriori dettagli sulle specifiche formali, fare riferimento a <http://www.rational.com> e <http://www.omg.org>.

I grafici UML sono costituiti dai seguenti elementi:

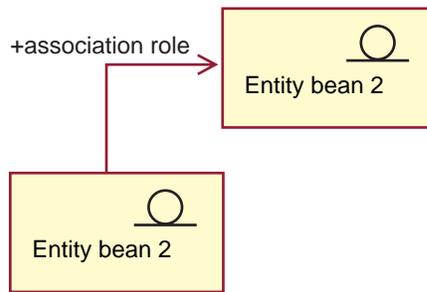
- Caselle: le caselle rappresentano classi di oggetti. I nomi delle classi vengono visualizzati nella parte alta della casella. Gli attributi, se sono visualizzati, sono disposti al di sotto del nome della classe. Il nome della classe e gli attributi sono separati da una linea.
- Linee: le linee rappresentano le possibili relazioni tra gli oggetti di due classi. Gli oggetti della classe a un'estremità della linea possono essere "associati" agli oggetti dell'altra classe.
- Figure romboidali piene: le figure romboidali piene all'estremità di una linea indicano ciò che viene contenuto per valore. Gli oggetti della classe all'altra estremità della linea fanno parte di un unico oggetto della classe toccata dal rombo.
- Figure romboidali vuote: le figure romboidali vuote all'estremità di una linea indicano ciò che viene contenuto per riferimento. Gli oggetti all'estremità della linea corrispondente al rombo possono essere considerati come raggruppamenti di oggetti della classe all'altra estremità della linea.
- Numeri cardinali: vengono visualizzati nella parte finale della linea delle relazioni per indicare una limitazione di cardinalità. La tabella seguente riepiloga le limitazioni di cardinalità:

Numero cardinale	Tipo di relazione
1	uno e solo uno
0..1	zero o uno
0..n	da zero in poi
1..n	da uno in poi

Se non è presente nessuna restrizione di cardinalità, si considera la cardinalità 0..n, a meno che non venga visualizzato un rombo pieno all'estremità di una linea delle relazioni. In questo caso, la cardinalità deve essere 1.

- Segni più: i segni più che vengono visualizzati nella parte finale delle linee delle relazioni indicano che l'oggetto della classe all'estremità della linea ha un ruolo nella relazione. Il testo che segue il segno più indica il ruolo dell'oggetto nella relazione.
- Frecce: le frecce all'estremità di una linea delle relazioni indicano la direzione della relazione tra due oggetti. La mancanza di frecce su una linea delle relazioni indica che la direzione della relazione tra gli oggetti è in entrambi i sensi.

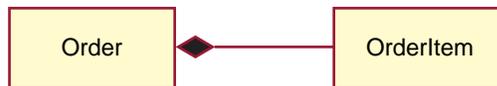
I grafici di seguito riportati illustrano i concetti sopra espressi:



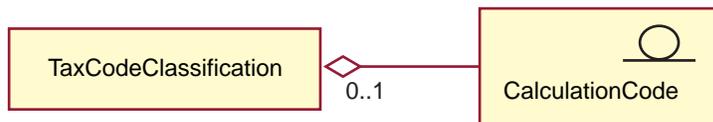
Questo diagramma mostra due bean di entità con il simbolo di stereotipo di decorazione che indica un Enterprise Java Bean. Esiste un'associazione unidirezionale dal primo bean al secondo bean di entità. Il segno più è seguito dal testo che descrive il ruolo che il bean di entità 2 ha nell'associazione.



In questo grafico un'entità di negozio ha un unico proprietario, che è un membro. Un membro può essere proprietario di zero o più entità secondarie. Il segno più indica che il membro ha un ruolo nella relazione. In questo caso, il membro è il proprietario dell'entità negozio. La freccia indica che è possibile trovare il proprietario di un'entità negozio chiedendo all'entità negozio il relativo proprietario e non chiedendo a un membro tutte le entità negozio che possiede.



In questo grafico un articolo dell'ordine è sempre parte di un unico ordine. Un ordine ha zero o più articoli di ordine.



Questo grafico indica che un codice di calcolo è raggruppato in base a zero o che una classificazione del codice di imposta raggruppa zero o più codici di calcolo.

Appendice B. Creazione di dati

Prima di creare i dati del negozio sotto forma di file XML, attenersi alla seguente procedura:

- Stabilire l'ordine delle informazioni che si desidera creare. Le informazioni presenti nei capitoli sui dati del negozio illustrano l'ordine in cui creare i dati, ma quando si creano file XML, le informazioni per una tabella parent devono precedere quelle per una tabella child.
- Stabilire il modo in cui si desidera utilizzare il negozio. Se si desidera creare un file dell'archivio negozio di esempio (.sar) vale a dire un archivio negozio da copiare e utilizzare come una base su cui creare nuovi negozi, è necessario creare dati leggermente diversi. Per ulteriori informazioni, consultare la sezione "Creazione di dati per i negozi di esempio".

Creazione di dati per i negozi di esempio

I dati contenuti negli archivi dei negozi di esempio assumono la forma di file XML ben strutturati validi per il pacchetto Loader. I file XML dell'archivio negozio sono concepiti per essere trasportati e non possono contenere chiavi primarie generate specifiche per una particolare istanza del database. Essi utilizzano invece alias interni che vengono risolti da ID Resolver al momento della pubblicazione. L'utilizzo di queste convenzioni permette agli archivi negozio di esempio di essere copiati e pubblicati più volte.



Non è necessario utilizzare queste convenzioni quando si creano dati per il negozio sotto forma di file XML, a meno che non si desideri creare un archivio negozio di esempio da utilizzare per creare altri negozi o un archivio negozio trasportabile, ovvero che può essere pubblicato in un'altra istanza di WebSphere Commerce.

Quindi, gli archivi negozio di esempio utilizzano le convenzioni di seguito riportate:

- & come in `member_id="&MEMBER_ID;"`, la convenzione `&XXX;` è una macro DTD (nota in XML come entità).

Nota: Quando si crea un archivio negozio di esempio, è necessario definire `MEMBER_ID` come una macro DTD.

WebSphere Commerce definisce un insieme di macro nei seguenti file:

-  unità: `\WebSphere\CommerceServer\xml\sar\DBLoadMacros.dtd`
-  unità: `\Program Files\WebSphere\CommerceServer\xml\sar\DBLoadMacros.dtd`
-  `/usr/WebSphere/CommerceServer/xml/sar/DBLoadMacros.dtd`
-  `/opt/WebSphere/CommerceServer/xml/sar/DBLoadMacros.dtd`
-  `/opt/WebSphere/CommerceServer/xml/sar/DBLoadMacros.dtd`
-  `/qibm/proddata/WebCommerce/xml/sar/DBLoadMacros.dtd`

Le macro `en_US` e `es_ES` vengono impostate sugli ID della lingua appropriata. Ad esempio:

```
<!ENTITY en_US "-1">
```

Le informazioni sono specificate utilizzando gli strumenti Servizi del negozio. Ad esempio, l'utente seleziona MEMBER_ID nella pagina di creazione dell'archivio negozio in Servizi del negozio. La macro MEMBER_ID è un segnaposto per l'ID del membro proprietario del negozio. Quando si crea un archivio negozio, si seleziona un membro che funga da proprietario del negozio. La macro MEMBER_ID viene impostata sull'ID di quel membro. Ad esempio, se si seleziona il membro ID -2000, allora MEMBER_ID viene impostato su -2000 nel modo seguente:

```
<!ENTITY MEMBER_ID "-2000">
```

- @ come in `ffmcenter_id="@ffmcenter_id_1"`. L'utilizzo del simbolo @ è riconosciuto come risoluzione dell'alias interno. ID Resolver, un programma di utilità del pacchetto Loader, genera identificativi per gli elementi XML che li richiedono. Una delle tecniche utilizzate da ID Resolver è la risoluzione alias interno. Quando si utilizzano le risoluzioni dell'alias interno, un alias sostituisce la chiave principale (identificativo) nel documento XML. L'alias viene quindi utilizzato altrove nel file XML per indicare tale elemento. Questo elimina la necessità di conoscere gli indici univoci necessari per creare il file XML. Durante la pubblicazione in Servizi del negozio o utilizzando il Pacchetto Loader, ID Resolver sostituisce il simbolo @ con un valore univoco. Fare riferimento agli esempi di seguito riportati di file XML:

– Pre-ID Resolver

```
<catalog
  catalog_id="@catalog_id_1"
  member_id="&MEMBER_ID;"
  identifier=InFashion"
  description="InFashion Catalog"/>
```

– Post ID Resolver

```
<catalog
  catalog_id="10001"
  member_id="-2000"
  identifier=InFashion"
  description="InFashion Catalog"/>
```

dove 10001 è l'ID univoco assegnato da ID Resolver e -2000 è l'ID del membro selezionato dall'utente in Servizi del negozio. Il file XML che ne deriva, viene caricato mediante il Pacchetto Loader. L'esecuzione dei file mediante ID Resolver garantisce la possibilità di creare numerosi negozi da un'unica serie di file XML.

Servizi del negozio e negozi di esempio

Le opzioni **Nuovo** e Crea archivio del negozio di Servizi del negozio dipendono dalle convenzioni descritte sopra. Se si desidera utilizzare il negozio come un modello per creare altri negozi mediante Servizi del negozio, è necessario seguire queste convenzioni quando si creano le risorse dati.

Tuttavia, se non si desidera creare un archivio negozio di esempio è possibile modificare o pubblicare un archivio negozio che non segua queste convenzioni utilizzando gli strumenti Servizi del negozio.

Appendice C. sarinfo.xml

Ogni archivio negozio deve includere un file `sarinfo.xml`. Questo file, noto come descrittore, contiene le informazioni sull'archivio negozio utilizzato quando viene pubblicato un archivio negozio, inclusi i nomi dei file ZIP dei file e i file XML del database del negozio e l'ordine in cui vengono pubblicati. Se un archivio negozio include i file in più lingue, il file `sarinfo.xml` include anche quelle informazioni e determina l'ordine in cui ogni file di lingua viene pubblicato.

Nota: L'ordine in cui vengono pubblicate le risorse dati è importante, dal momento che le risorse dati devono essere pubblicate prima delle altre risorse. Quindi, l'ordine delle risorse specificato nel file `sarinfo.xml`, deve corrispondere all'ordine delle risorse specificato nei file `sarinfo.xml` per i negozi di esempio. Gli archivi negozio di esempio si trovano nelle seguenti directory.

-  unità:\WebSphere\CommerceServer\samplestores
-  unità:\Program Files\WebSphere\CommerceServer\samplestores
-  /usr/WebSphere/CommerceServer/samplestores
-  /opt/WebSphere/CommerceServer/samplestores
-  /opt/WebSphere/CommerceServer/samplestores
-  /qibm/proddata/WebCommerce/samplestores

Per visualizzare il contenuto dell'archivio negozio, decomprimere il file con un programma ZIP. Il file `sarinfo.xml` si trova nella directory SAR-INF.

Esempio di sarinfo.xml

Di seguito è riportato il file `sarinfo.xml` ToolTech. Per ulteriori informazioni sugli elementi, gli attributi e i valori degli attributi, fare riferimento alle informazioni di seguito riportate. Per ulteriori informazioni sulle specifiche XML per un archivio di negozio, vedere il file `sarinfo.dtd` nella seguente directory:

-  unità:\WebSphere\CommerceServer\xml\sar
-  unità:\Program Files\WebSphere\CommerceServer\xml\sar
-  /usr/WebSphere/CommerceServer/xml/sar
-  /opt/WebSphere/CommerceServer/xml/sar
-  /opt/WebSphere/CommerceServer/xml/sar
-  /qibm/proddata/WebCommerce/xml/sar

```
<?xml version = "1.0"?>
<!DOCTYPE sarinfo SYSTEM "sarinfo.dtd">
<sarinfo complete-store="yes" multi-language="yes" version="1.0">

<store-info asset-name="store"/>

<file name="webapp.zip" type="zip">

<asset fragmented="no" name="webapp">
```

```

<file name="webapp.zip" type="zip">
<display-name>My Web App Display Name</display-name>
<description>My Web App</description>
</file>
</asset>

<asset fragmented="no" name="properties">
<file name="properties.zip" type="zip" />
</asset>

<asset fragmented="no" name="dbloadmacros">
<file name="data/DBLoadMacros.dtd" type="dtd"/>
</asset>

<asset fragmented="no" name="fulfillment">
<file name="data/fulfillment.dtd" type="dtd"/>
<file name="data/fulfillment.xml" priority="1" type="db-load"/>
<file name="data/en_US/fulfillment.xml" priority="31" type="db-load">
<locale>en_US</locale>
</file>
<file name="data/es_ES/fulfillment.xml" priority="31" type="db-load">
<locale>es_ES</locale>
</file>
</asset>

<asset fragmented="yes" name="store">
<file name="data/store.dtd" type="dtd"/>
<file name="data/store.xml" priority="2" type="db-load"/>
<file name="data/en_US/store.xml" priority="3" type="db-load">
<locale>en_US</locale>
</file>
<file name="data/es_ES/store.xml" priority="3" type="db-load">
<locale>es_ES</locale>
</file>
</asset>

<asset fragmented="yes" name="catalog">
<file name="data/catalog.dtd" type="dtd"/>
<file name="data/catalog.xml" priority="4" type="db-load"/>
<file name="data/en_US/catalog.xml" priority="5" type="db-load">
<locale>en_US</locale>
</file>
<file name="data/es_ES/catalog.xml" priority="5" type="db-load">
<locale>es_ES</locale>
</asset>

<asset fragmented="yes" name="tax">
<file name="data/tax.dtd" type="dtd"/>
<file name="data/tax.xml" priority="6" type="db-load"/>
<file name="data/en_US/tax.xml" priority="7" type="db-load">
<locale>en_US</locale>
</file>
<file name="data/es_ES/tax.xml" priority="7" type="db-load">
<locale>es_ES</locale>
</file>
</asset>

<asset fragmented="no" name="taxfulfill">
<file name="data/taxfulfill.dtd" type="dtd"/>
<file name="data/taxfulfill.xml" priority="8" type="db-load"/>
</asset>

<asset fragmented="yes" name="shipping">
<file name="data/shipping.dtd" type="dtd"/>
<file name="data/shipping.xml" priority="9" type="db-load"/>
<file name="data/en_US/shipping.xml" priority="10" type="db-load">
<locale>en_US</locale>

```

```

/file>
file name="data/es_ES/shipping.xml" priority="10" type="db-load">
<locale>es_ES</locale>
</file>
</asset>

<asset fragmented="no" name="shippingfulfill">
<file name="data/shipfulfill.dtd" type="dtd"/>
<file name="data/shipfulfill.xml" priority="11" type="db-load"/>
</asset>

<asset fragmented="no" name="store-catalog">
<file name="data/store-catalog.dtd" type="dtd"/>
<file name="data/store-catalog.xml" priority="12" type="db-load"/>
</asset>

<asset fragmented="no" name="storefulfill">
<file name="data/storefulfill.dtd" type="dtd"/>
<file name="data/storefulfill.xml" priority="13" type="db-load"/>
</asset>

<asset fragmented="yes" name="offering">
<file name="data/offering.dtd" type="dtd"/>
<file name="data/offering.xml" priority="14" type="db-load"/>
</asset>

<asset fragmented="no" name="command">
<file name="data/command.dtd" type="dtd"/>
<file name="data/command.xml" priority="16" type="db-load"/>
</asset>

<asset fragmented="yes" name="currency">
<file name="data/currency.dtd" type="dtd"/>
<file name="data/currency.xml" priority="17" type="db-load"/>
<file name="data/en_US/currency.xml" priority="18" type="db-load">
<locale>en_US</locale>
</file>
<file name="data/es_ES/currency.xml" priority="18" type="db-load">
<locale>es_ES</locale>
</file>
</asset>

<asset fragmented="yes" name="campaign">
<file name="data/campaign.dtd" type="dtd"/>
<file name="data/campaign.xml" priority="20" type="db-load"/>

<file name="data/en_US/campaign.xml" priority="24" type="db-load">
<locale>en_US</locale>
</file>
<file name="data/es_ES/campaign.xml" priority="24" type="db-load">
<locale>es_ES</locale>
</file>
</asset>

<asset fragmented="no" name="store-catalog-tax">
<file name="data/store-catalog-tax.dtd" type="dtd"/>
<file name="data/store-catalog-tax.xml" priority="21" type="db-load"/>
</asset>

<asset fragmented="no" name="store-catalog-shipping">
<file name="data/store-catalog-shipping.dtd" type="dtd"/>
<file name="data/store-catalog-shipping.xml" priority="22" type="db-load"/>
</asset>

<asset fragmented="no" name="store-defaults">
<file name="data/store-defaults.dtd" type="dtd"/>
<file name="data/store-defaults.xml" priority="23" type="db-load"/>

```

```

</asset>

<asset fragmented="no" name="consistency_check">
<file name="data/sarrule.dtd" type="dtd"/>
<file name="data/sarrule.xml" priority="25" type="config"/>
</asset>

<asset fragmented="no" name="payment">
<file name="data/es_ES/paymentinfo.xml" type="config"/>
<file name="data/paymentinfo.dtd" type="dtd"/>
</asset>

<asset fragmented="yes" name="policy">
<file name="data/businesspolicy.dtd" type="dtd"/>
<file name="data/businesspolicy.xml" priority="26" type="db-load"/>
<file name="data/en_US/businesspolicy.xml" priority="27" type="db-load">
<locale>en_US</locale>
</file>
<file name="data/es_ES/businesspolicy.xml" priority="27" type="db-load">
<locale>es_ES</locale>
</file>
</asset>

<asset fragmented="no" name="organization">
<file name="data/organization.dtd" type="dtd"/>
<file name="data/organization.xml" priority="28" type="db-load"/>
</asset>

asset fragmented="no" name="businessaccount">
  <file name="data/businessaccount.xml" type="xml"/>
</asset>
asset fragmented="yes" name="contract">
  <file name="data/contract.xml" priority="1" type="xml"/>
  <file name="data/en_US/contract.xml" priority="2" type="xml">
<locale>en_US</locale>
</file>
file name="data/es_ES/contract.xml" priority="2" type="xml">
<locale>es_ES</locale>
</file>
</asset>

<asset fragmented="yes" name="accesscontrol">
<file name="data/accesscontrol.dtd" type="dtd"/>
<file name="data/accesscontrol.xml" priority="29" type="db-load"/>
<file name="data/en_US/accesscontrol.xml" priority="30" type="db-load">
<locale>en_US</locale>
</file>
<file name="data/es_ES/accesscontrol.xml" priority="30" type="db-load">
<locale>es_ES</locale>
</file>
</asset>

<!-- next priority should be 32 -->

</sarinfo>

dove

```

sarinfo rappresenta le informazioni contenute nel file sarinfo.xml. Gli attributi del seguente grafico contengono informazioni generali sull'archivio negozio.

Nome attributo	Valore/i attributo/i
multi-language (obbligatorio)	Determina se in questo archivio negozio viene supportata l'opzione multilingua: <ul style="list-style-type: none"> • yes • no
complete-store (obbligatorio)	Determina se l'archivio negozio include le risorse necessarie per un negozio completo: <ul style="list-style-type: none"> • yes • no
version	La versione dell'archivio negozio. Ad esempio, 1.0, 1.1
display-name	Il nome dell'archivio negozio.
description	Una breve descrizione dell'archivio negozio.
standard-schema (obbligatorio)	L'archivio negozio segue lo schema di database standard di WebSphere Commerce: <ul style="list-style-type: none"> • yes • no
store-info asset-name (obbligatorio)	La risorsa che funziona come ancora per l'archivio negozio. Tutte le informazioni relative al negozio si trovano nei file che appartengono a questa risorsa. Ad esempio: store
locale	La locale supportata dall'archivio negozio. Le variabili delle locali (elencate di seguito) sono composte dalla lingua e dalla nazione. <ul style="list-style-type: none"> • de_DE • en_US • es_ES • fr_FR • it_IT • ja_JP • ko_KR • pt_BR • zh_CN • Zh_TW

asset (obbligatorio). Una risorsa è un insieme logico di file correlati. Ad esempio, *tassa* è il nome del gruppo di file correlati alle tasse del negozio.

Nome attributo	Valore/i attributo/i
name (obbligatorio)	Il nome del tipo di risorsa. Ad esempio: <ul style="list-style-type: none"> • store • catalog • payment • tax

Nome attributo	Valore/i attributo/i
fragmented (obbligatorio)	Determina se le informazioni sulle risorse vengono suddivise in più file a seconda delle lingue. <ul style="list-style-type: none"> • yes • no

file

Nome attributo	Valore/i attributo/i
name (obbligatorio)	Il nome del file.
type (obbligatorio)	Tipo di formato file. Ad esempio: <ul style="list-style-type: none"> • db-load - file da caricare nel database • dtd - file di definizione del tipo di documento • zip - file ZIP per le risorse del file, ad esempio webapp.zip • config - file di configurazione
priority	L'ordine in cui verranno pubblicati i file dell'archivio negozio. 1,2,3,4 . . . Nota: Se due file condividono la stessa priorità, l'ordine di caricamento è irrilevante. Se i file devono essere caricati in un determinato ordine, verificare che siano assegnate priorità diverse.
display-name	Il nome del file.
description	La descrizione di riferimento.
locale	La locale. Le variabili delle locali (elencate di seguito) sono composte dalla lingua e dalla nazione. <ul style="list-style-type: none"> • de_DE • en_US • es_ES • fr_FR • it_IT • ja_JP • ko_KR • pt_BR • zh_CN • Zh_TW

Appendice D. sarrule.xml

Ciascun archivio negozio contiene un file `sarrule.xml` che agisce come controllore della coerenza quando si esegue una pubblicazione mediante Servizi del negozio. Durante la pubblicazione, il programma di utilità per la pubblicazione utilizza le regole presenti nel file `sarrule.xml` per controllare che l'archivio negozio contenga le risorse Web elencate nei file XML.

Esempio di sarrule.xml

Il file `sarrule.xml` di seguito riportato si trova nel negozio di esempio ToolTech.

I file dell'archivio negozio si trovano nelle seguenti directory:

-  unità:\WebSphere\CommerceServer\samplstores
-  unità:\Program Files\WebSphere\CommerceServer\samplstores
-  /usr/WebSphere/CommerceServer/samplstores
-  /opt/WebSphere/CommerceServer/samplstores
-  /opt/WebSphere/CommerceServer/samplstores
-  /qibm/proddata/WebCommerce/samplstores

Per visualizzare il file `sarrule.xml` nell'archivio negozio, decomprimere il file con un programma ZIP. Il file `sarrule.xml` si trova nella directory `data`.

Il file `sarrule.dtd` si trova nella seguente directory:

-  unità:\WebSphere\CommerceServer\xml\sar
-  unità:\Program Files\WebSphere\CommerceServer\xml\sar
-  /usr/WebSphere/CommerceServer/xml/sar
-  /opt/WebSphere/CommerceServer/xml/sar
-  /opt/WebSphere/CommerceServer/xml/sar
-  /qibm/proddata/WebCommerce/xml/sar

Nota: Si consiglia di utilizzare il file `sarrule.xml` fornito con i negozi di esempio dell'archivio negozio. Tuttavia, se si desidera, è possibile aggiungere nuove regole al file `sarrule.xml` esistente.

```
<?xml version="1.0"?>
<!DOCTYPE SAR-rules SYSTEM "sarrule.dtd">
<SAR-rules>
  <asset name = "command">
    <check type="webasset registration">
      <rule entry="viewreg" attribute="properties" type="java.lang.String"
        removeStoreDir="false" file="webapp.zip"/>
    </check>
  </asset>
!--
  <asset name = "catalog">
    <check type="webasset registration">
      <rule entry = "catalogdsc" attribute="thumbnail" type="java.lang.String"
        removeStoreDir="false" file="webapp.zip"/>
    </check>
  </asset>
</SAR-rules>
```

```

<rule entry = "catalogdsc" attribute="fullimage" type="java.lang.String"
  removeStoreDir="false" file="webapp.zip"/>
<rule entry = "catentdesc" attribute="thumbnail" type="java.lang.String"
  removeStoreDir="false" file="webapp.zip"/>
<rule entry = "catentdesc" attribute="fullimage" type="java.lang.String"
  removeStoreDir="false" file="webapp.zip"/>
<rule entry = "catgrpdesc" attribute="thumbnail" type="java.lang.String"
  removeStoreDir="false" file="webapp.zip"/>
<rule entry = "catgrpdesc" attribute="fullimage" type="java.lang.String"
  removeStoreDir="false" file="webapp.zip"/>
</check>
</asset>
-->
<asset name = "store">
<check type="webasset registration">
  <rule entry = "dispentrel" attribute="pagename" type="java.lang.String"
    removeStoreDir="false" file="webapp.zip"/>
  <rule entry = "dispcgprel" attribute="pagename" type="java.lang.String"
    removeStoreDir="false" file="webapp.zip"/>
</check>
</asset>
</SAR-rules>

```

dove

```

<asset name = "command">
<check type="webasset registration">
  <rule entry="viewreg" attribute="properties" type="java.lang.String"
    removeStoreDir="false" file="webapp.zip"/>
</check>
</asset>

```

Il programma di utilità per la pubblicazione di Servizi del negozio verifica che ciascun file JSP visualizzato nel file command.xml sia presente nelle risorse Web dell'archivio negozio.

dove

```

<asset name = "catalog">
<check type="webasset registration">
  <rule entry = "catalogdsc" attribute="thumbnail" type="java.lang.String"
    removeStoreDir="false" file="webapp.zip"/>
  <rule entry = "catalogdsc" attribute="fullimage" type="java.lang.String"
    removeStoreDir="false" file="webapp.zip"/>
  <rule entry = "catentdesc" attribute="thumbnail" type="java.lang.String"
    removeStoreDir="false" file="webapp.zip"/>
  <rule entry = "catentdesc" attribute="fullimage" type="java.lang.String"
    removeStoreDir="false" file="webapp.zip"/>
  <rule entry = "catgrpdesc" attribute="thumbnail" type="java.lang.String"
    removeStoreDir="false" file="webapp.zip"/>
  <rule entry = "catgrpdesc" attribute="fullimage" type="java.lang.String"
    removeStoreDir="false" file="webapp.zip"/>
</check>
</asset>

```

Il programma di utilità per la pubblicazione di Servizi del negozio verifica che ciascuna risorsa catalogo visualizzata nel file catalog.xml sia presente nelle risorse Web dell'archivio negozio.

dove

```

<asset name = "store">
<check type="webasset registration">
  <rule entry = "dispentrel" attribute="pagename" type="java.lang.String"
    removeStoreDir="false" file="webapp.zip"/>

```

```
<rule entry = "dispcgpre1" attribute="pagename" type="java.lang.String"
      removeStoreDir="false" file="webapp.zip"/>
</check>
</asset>
```

Il programma di utilità per la pubblicazione di Servizi del negozio verifica che ciascuna risorsa del negozio visualizzata nel file `store.xml` sia presente nelle risorse Web dell'archivio negozio.



Se la memoria è un problema durante la pubblicazione, commentare la regola della risorsa catalogo nel file `sarrule.xml` prima della pubblicazione.

Appendice E. Gruppi di risorse database

Tutte le risorse database di WebSphere Commerce sono suddivise in gruppi per la creazione e il caricamento. Questi gruppi costituiscono un insieme collegato logicamente di tabelle. L'ordine in cui sono organizzati i gruppi di risorse database è importante per il caricamento dei dati perché alcuni oggetti devono esistere per caricare la relazione tra gli oggetti.

Quando le risorse database vengono caricate per il negozio in formato XML, è possibile scegliere solo gruppi selezionati. Questi gruppi costituiscono le risorse database create nei precedenti capitoli, ad esempio catalogo o evasione ordini. Prima di caricare i gruppi di dati come descritto in "Caricamento dei gruppi di risorse database" a pagina 279, effettuare le seguenti operazioni:

- Decidere quale gruppo di risorse database caricare. Ogni gruppo contiene dipendenze che devono essere rispettate prima di poter caricare le risorse. Fare riferimento alle informazioni nell'"Dipendenze di gruppi di risorse database".
- Assicurarsi di aver creato o aggiornato i file XML per il gruppo di risorse selezionato. Le informazioni presenti nei capitoli sulle risorse illustrano l'ordine in cui creare le risorse database, ma quando si creano o si aggiornano file XML, le informazioni per una tabella parent devono precedere quelle per una tabella child.

Dipendenze di gruppi di risorse database

Ciascun gruppo di risorse database ricava le informazioni dalle tabelle di database di WebSphere Commerce. Le risorse database hanno dipendenze all'interno del relativo gruppo. In altre parole, un gruppo di risorse database non può ricavare i dati in altri file XML di un gruppo dati diverso e ciascun gruppo è indipendente meno le chiavi esterne. Tuttavia, se il gruppo di risorse database deve fare riferimento ai dati esterni definiti in un altro gruppo, è necessario fornire i dati manualmente. Ciò significa che i dati di un gruppo hanno una *dipendenza esterna* dai dati definiti al di fuori del dominio corrispondente, cioè da un altro gruppo di risorse database. Le dipendenze esterne ricorrono quando un gruppo di risorse database ha una relazione di chiave esterna con la chiave primaria di una tabella di un altro gruppo. Per caricare il gruppo di risorse database, è necessario soddisfare le dipendenze esterne. Per utilizzare un esempio dal grafico di seguito riportato, una delle dipendenze esterne del gruppo di risorse database del negozio è `fulfillment.FFMCENTER.FFMCENTER_ID`, che indica che le risorse database di evasione ordini devono esistere nel database WebSphere Commerce prima di poter caricare il gruppo di risorse database del negozio.

Prima di cominciare il processo di caricamento, osservare il grafico di seguito riportato. Ciascun gruppo di risorse database dipende da altre tabelle di database dalle quali vengono caricati i dati.

Alcuni punti da ricordare:

- Alcune dipendenze esterne potrebbero non essere soddisfatte da un solo gruppo. Le risorse database generali o a livello del sito utilizzate da ogni negozio sono già riempite al momento della creazione dell'istanza nel *bootstrap* e sono immediatamente accessibili. Le tabelle contenute nei gruppi di risorse database hanno riferimenti di chiavi esterne per questo tipo di dati. I dati di bootstrap sono suddivisi in dati comuni e dati specifici della locale. Se si dispone di un

negozio multilingue, è necessario scegliere sia i dati di bootstrap comuni che quelli specifici della locale. Ad esempio, sono necessari i dati di bootstrap relativi al membro e alla lingua. Il processo di creazione dell'istanza inserisce nella tabella LANGUAGE le lingue supportate da WebSphere Commerce per il negozio e crea un'organizzazione principale (MEMBER.MEMBER_ID=-2001) e un'organizzazione predefinita (MEMBER.MEMBER_ID=-2000). È necessario utilizzare l'organizzazione principale quando richiesto, ma occorre creare un'organizzazione proprietaria del negozio invece di utilizzare l'organizzazione predefinita. Per ulteriori informazioni sulle organizzazioni e la relativa gerarchia, fare riferimento alla guida in linea di WebSphere Commerce.

- I file elencati nella colonna delle **dipendenze esterne** utilizzano la seguente struttura di denominazione: *gruppo risorse database.tabella database.colonna database*. Utilizzando il file *store.STOREENT.STOREENT_ID* come esempio, i dati vengono ricavati dal gruppo di risorse *negozio*, dalla tabella *STOREENT* e dalla colonna *STOREENT_ID*. I nomi file che cominciano con *bootstrap* indicano che i dati sono stati riempiti durante la creazione dell'istanza di WebSphere Commerce.
- I file elencati nella colonna delle **dipendenze esterne** contengono i riferimenti delle chiavi esterne alle **tabelle correlate**. Queste tabelle devono essere riempite per prime.
- Per motivi di presentazione, le tabelle sono state divise in modo da indicare quelle specifiche della locale che contengono le informazioni su più lingue, ad esempio le descrizioni dei prodotti.
- Le tabelle del grafico rappresentano le risorse database dei negozi di esempio di WebSphere Commerce. Le tabelle possono variare a seconda delle esigenze, delle funzioni e delle dimensioni del negozio. A seconda dei requisiti del negozio, assicurarsi di includere tutte le tabelle di database contenenti risorse negozio, anche quelle non elencate di seguito.

Risorse database di controllo accessi		
Dipendenze esterne	Tabelle correlate dei file XML di risorse database	Tabelle correlate specifiche della locale dei file XML di risorse database
bootstrap.LANGUAGE.LANGUAGE_ID (organizzazione principale e di proprietario negozio), bootstrap.MEMBER.MEMBER_ID (organizzazione principale e di proprietario negozio)	accesscontrol.xml ACACTACTGP, ACACTGRP, ACACTION, ACPOLICY, ACRESCGRY, ACRESGPRES, ACRESGRP	accesscontrol.xml ACACGPDESC, ACACTDESC, ACPOLDESC, ACRSCGDES, ACRESGPDES
Risorse database delle politiche commerciali		
Dipendenze esterne	Tabelle correlate dei file XML di risorse database	Tabelle correlate specifiche della locale dei file XML di risorse database
bootstrap.LANGUAGE.LANGUAGE_ID, bootstrap.MEMBER.MEMBER_ID, store.STOREENT.STOREENT_ID (organizzazione proprietaria del negozio)	businesspolicy.xml POLICY, POLICYCMD	businesspolicy.xml POLICYDESC
Risorse database delle campagne		

Dipendenze esterne	Tabelle correlate dei file XML di risorse database	Tabelle correlate specifiche della locale dei file XML di risorse database
store.STOREENT.STOREENT_ID	campaign.xml CAMPAIGN, COLLATERAL, EMSPOT, STENCALUSG	campaign.xml COLLDESC
Risorse database di catalogo		
Dipendenze esterne	Tabelle correlate dei file XML di risorse database	Tabelle correlate specifiche della locale dei file XML di risorse database
bootstrap.LANGUAGE.LANGUAGE_ID, bootstrap.MEMBER.MEMBER_ID (organizzazione proprietaria del negozio), store.STOREENT.STOREENT_ID, shipping.CALCODE.CALCODE_ID, tax.CALCODE.CALCODE_ID	catalog.xml BASEITEM, CATALOG, CATENTREL, CATENTRY, CATGROUP, CATGRPREL, CATTOGRP, ITEMSPC, ITEMVERSN, RA, RADETAIL, STOREITEM, STORITMFFC, VERSIONSPC offering.xml CATGRPTPC, MGPTRDPSCN, OFFER, OFFERPRICE, TRADEPOSCN storefulfill.xml INVENTORY store-catalog.xml DISPCGPREL, DISPENTREL, STORECAT, STORECENT, STORECGRP store-catalog- shipping.xml CATENTCALCD, CATENTSHIP store-catalog- tax.xml CATENTCALD	catalog.xml ATTRIBUTE, ATTRVALUE, BASEITMDSC, CATALOGDSC, CATENTDESC, CATGRPDESC, PKGATTR, PKGATTRVAL,
Risorse database dei comandi		
Dipendenze esterne	Tabelle correlate dei file XML di risorse database	Tabelle correlate specifiche della locale dei file XML di risorse database
store.STOREENT.STOREENT_ID	command.xml CMDREG, VIEWREG	N/A

Business Risorse database dei contratti		
Dipendenze esterne	Tabelle correlate dei file XML di risorse database	
store.STOREENT.STOREENT_ID	Le tabelle di database dei contratti non vengono caricate direttamente e seguono un processo diverso rispetto agli altri gruppi di dati di WebSphere Commerce. Per ulteriori informazioni, fare riferimento a "Pubblicazione delle risorse contratto" a pagina 287.	
Risorse database delle valute		
Dipendenze esterne	Tabelle correlate dei file XML di risorse database	Tabelle correlate specifiche della locale dei file XML di risorse database
store.STOREENT.STOREENT_ID	currency.xml CURCVLIST	currency.xml CURCONVERT, CURLIST
Risorse database di evasione ordini		
Dipendenze esterne	Tabelle correlate dei file XML di risorse database	Tabelle correlate specifiche della locale dei file XML di risorse database
bootstrap.LANGUAGE.LANGUAGE_ID, bootstrap.MEMBER.MEMBER_ID (organizzazione proprietaria del negozio)	fulfillment.xml FFMCENTER, STADDRESS	fulfillment.xml FFMCENTDS
Risorse database dell'organizzazione		
Dipendenze esterne	Tabelle correlate dei file XML di risorse database	Tabelle correlate specifiche della locale dei file XML di risorse database
bootstrap.LANGUAGE.LANGUAGE_ID (organizzazione principale e di proprietario negozio), bootstrap.MEMBER.MEMBER_ID (organizzazione principale e di proprietario negozio)	organization.xml ADDRBOOK, ADDRESS, MBRREL, MEMBER, ORGENITY	N/A
Risorse database delle spedizioni		
Dipendenze esterne	Tabelle correlate dei file XML di risorse database	Tabelle correlate specifiche della locale dei file XML di risorse database

bootstrap.LANGUAGE.LANGUAGE_ID, bootstrap.MEMBER.MEMBER_ID (organizzazione proprietario negozio), fulfillment.FFMCENTER.FFMCENTER_ID, store.STOREENT.STOREENT_ID	shipping.xml CALCODE, CALRULE, CRULESCALE, JURST, JURSTGPREL, JURSTGROUP, SHIPMODE, STENCALUSG shipping.xml SHPJCRULE, SHPARRANGE	shipping.xml CALCODEDSC, CALRANGE, CALRLOOKUP, CALSCALE, SHPMODEDSC
Risorse database di negozio		
Dipendenze esterne	Tablelle correlate dei file XML di risorse database	Tablelle correlate specifiche della locale dei file XML di risorse database
bootstrap.LANGUAGE.LANGUAGE_ID, bootstrap.MEMBER.MEMBER_ID (organizzazione proprietaria del negozio), bootstrap.SETCURR.SETCURR_ID, fulfillment.FFMCENTER.FFMCENTER_ID	store.xml INVADJCODE, RTNREASON, STORE, STOREENT, STORELANG, VENDOR	store.xml FFMCENTDS, INVADJDESC, RTNRSNDESC, STADDRESS, STOREENTDS, STORLANGDS, VENDORDESC
Risorse database predefinito del negozio		
Dipendenze esterne	Tablelle correlate dei file XML di risorse database	Tablelle correlate specifiche della locale dei file XML di risorse database
shipping.SHIPMODE.SHIPMODE_ID (se applicabile, questo file deve essere caricato per primo), contract.CONTRACT.CONTRACT_ID, store.STOREENT.STOREENT_ID	store-default.xml STOREDEF	N/A
Risorse database delle tasse		
Dipendenze esterne	Tablelle correlate dei file XML di risorse database	Tablelle correlate specifiche della locale dei file XML di risorse database
bootstrap.LANGUAGE.LANGUAGE_ID, bootstrap.MEMBER.MEMBER_ID (organizzazione proprietaria del negozio), store.STOREENT.STOREENT_ID, fulfillment.FFMCENTER.FFMCENTER_ID, store.STOREENT.STOREENT_ID	tax.xml CALCODE, CALRANGE, CALRLOOKUP, CALRULE, CALSCALE, CRULESCALE, JURST, JURSTGROUP, JURSTGPREL, STENCALUSG, TAXCGRY, TAXJCRULE taxfulfill.xml TAXJCRULE	tax.xml CALCODEDSC, CALSCALEDS, TAXCGRYDS

Appendice F. Informazioni particolari

Queste informazioni sono state sviluppate per i prodotti e i servizi offerti negli Stati Uniti. È possibile che negli altri paesi l'IBM non offra i prodotti, i servizi o le funzioni illustrati in questo documento. Per informazioni sui prodotti o i servizi disponibili nella propria zona, contattare il rappresentante IBM locale. Qualsiasi riferimento a prodotti, programmi o servizi IBM non implica che l'IBM intenda renderli disponibili in tutti i paesi in cui opera. In sostituzione a quelli forniti dall'IBM possono essere usati prodotti, programmi o servizi funzionalmente equivalenti che non comportino violazione dei diritti di proprietà intellettuale dell'IBM. E' comunque responsabilità dell'utente valutare e verificare la possibilità di utilizzare altri programmi e/o prodotti, fatta eccezione per quelli espressamente indicati dall'IBM.

L'IBM può avere brevetti o domande di brevetto in corso relativi a quanto trattato nella presente pubblicazione. La fornitura di questa pubblicazione non implica la concessione di alcuna licenza su di essi. Per eventuali domande sulle licenze, scrivere al seguente indirizzo:

IBM Director of Licensing
IBM Europe
Schoenaicher Str. 220
D-7030 Boeblingen
Deutschland

Per richieste di informazioni relative alla licenza riguardanti informazioni double-byte (DBCS), fare richiesta scrivendo a:

IBM World Trade Asia Corporation
Licensing
2-31 Roppongi 3-chome, Minato-ku
Tokyo 106, Japan

Il seguente paragrafo non è valido per il Regno Unito o per tutti i paesi le cui leggi nazionali siano in contrasto con le disposizioni in esso contenute:

L'INTERNATIONAL BUSINESS MACHINES CORPORATION FORNISCE QUESTA PUBBLICAZIONE "NELLO STATO IN CUI SI TROVA", SENZA ALCUNA GARANZIA, ESPLICITA O IMPLICITA, IVI INCLUSE EVENTUALI GARANZIE DI COMMERCIALIZZABILITÀ ED IDONEITÀ AD UNO SCOPO PARTICOLARE. Alcuni stati non consentono la rinuncia a garanzie esplicite o implicite in determinate transazioni; quindi la presente dichiarazione potrebbe non essere a voi applicabile.

Questa pubblicazione potrebbe contenere imprecisioni tecniche o errori tipografici. Le informazioni incluse in questo documento vengono modificate su base periodica; tali modifiche verranno incorporate nelle nuove edizioni della pubblicazione. L'IBM si riserva il diritto di apportare miglioramenti e/o modifiche al prodotto o al programma descritto nel manuale in qualsiasi momento e senza preavviso.

Tutti i riferimenti a siti Web non dell'IBM contenuti in questo documento sono forniti solo per consultazione. I materiali disponibile presso i siti Web non fanno parte di questo prodotto e l'utilizzo di questi è a discrezione dell'utente.

Tutti i commenti e i suggerimenti inviati potranno essere utilizzati liberamente dall'IBM e dalla Selfin e diventeranno esclusiva delle stesse.

Coloro che detengono la licenza su questo programma e desiderano avere informazioni su di esso allo scopo di consentire (i) uno scambio di informazioni tra programmi indipendenti ed altri (compreso questo) e (ii) l'uso reciproco di tali informazioni, dovrebbero rivolgersi a:

IBM Canada Ltd.
Office of the Lab Director
8200 Warden Avenue Markham, Ontario L6G 1C7
Canada

Queste informazioni possono essere rese disponibili secondo condizioni contrattuali appropriate, compreso, in alcuni casi, il pagamento di un addebito.

Il programma su licenza descritto in questo manuale e tutto il materiale su licenza ad esso relativo sono forniti dall'IBM nel rispetto delle condizioni previste dalla licenza d'uso.

Tutti i dati relativi alle prestazioni contenuti in questa pubblicazione sono stati determinati in un ambiente controllato. Pertanto, i risultati ottenuti in ambienti operativi diversi possono variare in modo considerevole. Alcune misure potrebbero essere state fatte su sistemi di livello di sviluppo per cui non si garantisce che queste saranno uguali su tutti i sistemi disponibili. Inoltre, alcune misure potrebbero essere state ricavate mediante estrapolazione. I risultati possono quindi variare. Gli utenti di questa pubblicazione devono verificare che i dati siano applicabili al loro specifico ambiente.

Le informazioni relative a prodotti non IBM sono state ottenute dai fornitori di tali prodotti. L'IBM non ha verificato tali prodotti e, pertanto, non può garantirne l'accuratezza delle prestazioni. Eventuali commenti relativi alle prestazioni dei prodotti non IBM devono essere indirizzati ai fornitori di tali prodotti.

Tutte le dichiarazioni riguardanti la futura direzione o le intenzioni della IBM sono soggette a sostituzione o al ritiro senza preavviso, e rappresentano unicamente scopi e obiettivi della IBM stessa.

Tutti i prezzi IBM riportati sono prezzi al dettaglio suggeriti dalla IBM stessa, sono correnti e soggetti a cambiamenti senza preavviso. I prezzi al fornitore possono variare.

Queste informazioni hanno solo scopo di pianificazione. Queste informazioni possono essere soggette a variazioni prima che i prodotti descritti siano disponibili.

Questa pubblicazione contiene esempi di dati e prospetti utilizzati quotidianamente nelle operazioni aziendali, pertanto, può contenere nomi di persone, società, marchi e prodotti. Tutti i nomi contenuti nel manuale sono fittizi e ogni riferimento a nomi ed indirizzi reali è puramente casuale.

LICENZA SOGGETTA ALLE LEGGI SUL DIRITTO D'AUTORE:

Queste informazioni contengono esempi di programmi applicativi in lingua originale, che illustrano le tecniche di programmazione su diverse piattaforme operative. Potete copiare, modificare e distribuire questi esempi di programmi sotto qualsiasi forma senza alcun pagamento alla IBM, allo scopo di sviluppare, utilizzare, commercializzare o distribuire i programmi applicativi in modo conforme alle API (Application Programming Interface) a seconda della piattaforma operativa per cui gli esempi dei programmi sono stati scritti. Questi esempi non sono stati testati approfonditamente tenendo conto di tutte le condizioni possibili. La IBM, quindi, non può garantire o assicurare la affidabilità, la praticità o il funzionamento di questi programmi. Potete copiare, modificare e distribuire questi esempi di programmi sotto qualsiasi forma senza alcun pagamento alla IBM, allo scopo di sviluppare, utilizzare, commercializzare o distribuire i programmi applicativi in modo conforme alle API (Application Programming Interface) IBM.

Ogni copia o parte di tali programmi di esempio deve includere un'informazione sul copyright come questa di seguito indicata:

©Copyright International Business Machines Corporation 2001. Parti di questo programma derivano da programmi di esempio della IBM Corp. Programmi di esempio. ©Copyright IBM Corp. 2000, 2001. Tutti i diritti riservati.

Se questa pubblicazione viene visualizzata in formato elettronico, è possibile che le fotografie e le illustrazioni a colori non vengano visualizzate.

Le immagini di carte di credito e i nomi di marchi forniti con questo prodotto devono essere utilizzati solo dai rivenditori autorizzati dalla società fornitrice della carta di credito ad accettare i pagamenti effettuati mediante carta di credito.

Marchi

I seguenti termini sono marchi dell'IBM Corporation:

AIX	IBM	WebSphere
AS/400	IBM Payment Manager	
DB2	iSeries	
DB2 Universal Database	OS/400	
eServer	VisualAge	

Microsoft, Windows e Windows NT, Active Directory e il logo di Windows sono marchi o marchi registrati della Microsoft Corporation negli Stati Uniti e/o in altri paesi.

Oracle è un marchio registrato e Oracle8 è un marchio della Oracle Corporation.

SET Secure Electronic Transaction, SET e il logo di SET sono marchi di proprietà della SET Secure Electronic Transaction LLC. È vietato l'utilizzo di questi marchi senza una licenza scritta della SET Secure Electronic Transaction LLC.

Solaris, Solaris Operating Environment, Java, JavaServer Pages, JavaBeans e tutti i marchi basati su Java e i logo sono marchi o marchi registrati della Sun Microsystems, Inc.

UNIX è un marchio registrato di The Open Group negli Stati Uniti e in altri paesi.

Altri nomi di prodotti, società o servizi potrebbero essere marchi o marchi di servizi di altre società.

IBM