

IBM Content Manager OnDemand for iSeries Common
Server



Web Enablement Kit

Průvodce instalací a konfigurací

Verze 5, Vydání 3

IBM Content Manager OnDemand for iSeries Common
Server



Web Enablement Kit

Průvodce instalací a konfigurací

Verze 5, Vydání 3

Poznámka

Dříve, než použijete tyto informace a odpovídající produkt, přečtěte si informace uvedené v části “Poznámky” na stránce 139.

Třetí vydání (Květen 2004)

Toto vydání se vztahuje na produkt IBM Content Manager OnDemand for iSeries Common Server, verze 5, vydání 3, a na všechna následující vydání a modifikace, pokud v nových vydáních nebude uvedeno jinak. Toto vydání nahrazuje vydání SC27-1163-01.

© Copyright International Business Machines Corporation 2001, 2004. Všechna práva vyhrazena.

Obsah

O knize IBM Content Manager OnDemand for iSeries Common Server Web Enablement Kit - Průvodce instalací a konfigurací (SC27-1163) v

Kdo by měl číst tuto knihu.	v
Jak je tato kniha uspořádána	v
Nezbytné předpoklady a související informace	v
Jiné informace dostupné v síti WWW	vi
iSeries Navigator	vi
Jak zasílat připomínky.	vi

| Přehled změn vii

Kapitola 1. Přehled 1

O programovacích rozhraních	2
O prohlížečích	4
Použití ODWEK	5
Funkce produktu	5
Funkce Add Annotation (Přidání anotace)	5
Funkce Change Password (Změna hesla)	5
Funkce Document Hit List (Seznam nalezených dokumentů)	5
Funkce Logoff (Odhlášení)	6
Funkce Logon (Přihlášení).	6
Funkce Retrieve Document (Načtení dokumentu)	6
Funkce Search Criteria (Vyhledávací kritéria)	6
Funkce Server Print Document (Tisk dokumentů na serveru)	6
Funkce Update Document (Aktualizace dokumentu)	6
Funkce View Annotations (Zobrazení anotací).	6
Zabezpečení serveru a dat.	6

Kapitola 2. Instalace a konfigurace HTTP serveru 9

Požadavky na instalaci	9
Jiné požadavky	9
Instalace v OS/400	10
Váš další krok	10
Zadání souboru ARSWWW.INI.	10
[@SRV@_DEFAULT]	11
[@SRV@_server]	12
[CONFIGURATION].	13
[SECURITY]	19
[AFP2HTML]	20
[AFP2PDF].	21
[MIMETYPES]	22
[ATTACHMENT IMAGES].	26
[NO HTML]	28
[DEFAULT BROWSER].	28
[browser]	35
[DEBUG]	36
Příklad souboru ARSWWW.INI.	36
Váš další krok	38

Kapitola 3. Konfigurace vzorových aplikací 39

LOGON.HTM	39
CREDIT.HTM	39
TEMPLATE.HTM	40
Váš další krok	40

Kapitola 4. Instalace webových prohlížečů 41

Přehled	41
Požadavky	42
Instalace.	42
Distribuce uživatelsky definovaných souborů.	43
Instalace souborů AFP Web Viewer	44
Přidávání podadresářů	44
Ukládání uživatelsky definovaných souborů	45
Konfigurace souborů fontů	45
Vytvoření instalačního souboru AFP Web Viewer	46
Instalace softwaru AFP Web Viewer na pracovní stanici uživatele	46
Mapování fontů AFP	46
AFP Web Viewer	47
Image Web Viewer	47
Java prohlížeč Line Data Viewer	48
Váš další krok	51

Dodatek A. Referenční informace pro rozhraní CGI API 53

Funkce Add Annotation (Přidání anotace).	54
Funkce Change Password (Změna hesla)	57
Funkce Document Hit List (Seznam nalezených dokumentů).	59
Funkce Logoff (Odhlášení)	63
Funkce Logon (Přihlášení)	65
Funkce Print Document (Server) (Tisk dokumentu)	67
Funkce Retrieve Document (Načtení dokumentu)	71
Funkce Search Criteria (Vyhledávací kritéria)	74
Funkce Update Document (Aktualizace dokumentu)	76
Funkce Zobrazení anotací (View Annotations)	78

| Dodatek B. Reference pro Java servlet 81

| Dodatek C. Reference pro rozhraní Java API 83

| Dodatek D. Průvodce programováním rozhraní Java API 85

Architektura klient/server	85
Balení pro prostředí Java	85
Rady k programování	86
Nastavení systémové proměnné.	86
Nastavení proměnných prostředí	86
Trasování a diagnostické informace	88

	Trasování	88
	Zacházení s výjimkami	88
	Konstanty	88
	Spuštění aplikace ODWEK	89
	Připojování k serveru OnDemand	89
	Vytvoření připojení	89
	Nastavení a získání hesel	90
	Práce se serverem OnDemand	90
	Výpis skupin aplikací ve složce	92
	Hledání složky	94
	Vyhledání složky pomocí řetězce SQL	98
	Zrušení hledání	100
	Výpis kritérií hledání	102
	Výpis složek a informací o složkách	105
	Zobrazení seznamu dokumentů	107
	Načtení dokumentu	109
	Tisk dokumentu	112
	Výpis informací o poznámkách	113
	Přidání poznámky	115
	Aktualizace dokumentu	117
	Změna hesla	120

Dodatek E. Konverze AFP na HTML 123

Formát souboru AFP2HTML.INI	123
Volby pro službu AFP2WEB	124
Prohlížení konvertovaných dokumentů	125

Dodatek F. Konverze AFP na PDF 127

Zadání souboru AFP2PDF.INI	127
Prohlížení konvertovaných dokumentů	128

Dodatek G. Konfigurační soubory HTTP serveru 129

HTTP Apache Server	129
WebSphere Application Server	130

Dodatek H. Žádný HTML výstup 131

Výstup ASCII s oddělovači	131
Přihlášení	131
Poznámky	132
Vyhledávací kritéria	132
Poznámky	132
Seznam nalezených dokumentů	133
Poznámky	133
Zobrazení anotací	134
Chybová zpráva	134
Poznámky	134

Dodatek I. Podpora národního jazyka 135

Dodatek J. Nástroje k určování problémů 137

Poznámky 139	
Ochranné známky a servisní značky	140

Rejstřík 143

O knize IBM Content Manager OnDemand for iSeries Common Server Web Enablement Kit - Průvodce instalací a konfigurací (SC27-1163)

Tato kniha uvádí informace, které můžete použít k plánování, instalaci, konfiguraci a použití produktu IBM Content Manager OnDemand for iSeries, verze 5, vydání 3, Common Server (OnDemand) Web Enablement Kit.

Kdo by měl číst tuto knihu

Tato kniha je určena hlavně pro systémové administrátory, kteří potřebují implementovat, instalovat a udržovat software a aplikace OnDemand Web Enablement Kit (ODWEK). Také ji mohou používat programátoři, kteří potřebují integrovat server OnDemand s webovými aplikacemi.

Jak je tato kniha uspořádána

Tato kniha uvádí informace, které potřebujete k instalaci a konfiguraci aplikace ODWEK a k naplánování přístupu uživatelů k datům ze systému IBM Content Manager OnDemand for iSeries Common Server s webovým prohlížečem. Tato publikace obsahuje následující sekce:

- Kapitola 1, “Přehled”, na stránce 1
- Kapitola 2, “Instalace a konfigurace HTTP serveru”, na stránce 9
- Kapitola 3, “Konfigurace vzorových aplikací”, na stránce 39
- Kapitola 4, “Instalace webových prohlížečů”, na stránce 41
- Dodatek A, “Referenční informace pro rozhraní CGI API”, na stránce 53
- Dodatek B, “Reference pro Java servlet”, na stránce 81
- Dodatek C, “Reference pro rozhraní Java API”, na stránce 83
- Dodatek D, “Průvodce programováním rozhraní Java API”, na stránce 85
- Dodatek E, “Konverze AFP na HTML”, na stránce 123
- Dodatek F, “Konverze AFP na PDF”, na stránce 127
- Dodatek G, “Konfigurační soubory HTTP serveru”, na stránce 129
- Dodatek H, “Žádný HTML výstup”, na stránce 131
- Dodatek I, “Podpora národního jazyka”, na stránce 135
- Dodatek J, “Nástroje k určování problémů”, na stránce 137

Nezbytné předpoklady a související informace

Jako výchozí bod pro vyhledávání technických informací o serveru iSeries použijte aplikaci IBM iSeries Information Center.

Přístup k aplikaci Information Center je možný dvěma způsoby:

- Z webových stránek: <http://www.ibm.com/eserver/series/infocenter>
- Z disků CD-ROM, které jste obdrželi s licencovaným programem Operating System/400: *iSeries Information Center*, SK3T-7436-04. Tento balík programů také zahrnuje verze PDF publikací iSeries v sadě s názvem *Aplikace iSeries Information Center: Doplnkové manuály*, SK3T-7437-01. Tento disk CD-ROM nahrazuje CD-ROM se softcopy knihovnou.

Aplikace Information Center obsahuje pomocné programy a důležitá témata, jako např. Java, TCP/IP, webové služby, zabezpečené sítě, logické části systému, klastrování, CL příkazy a systémová rozhraní API. Rovněž zahrnuje odkazy na související červené knihy IBM a internetové odkazy na jiné webové stránky IBM, např. na domovskou stránku IBM.

Jiné informace dostupné v síti WWW

Další informace o serveru iSeries jsou k dispozici na WWW. Můžete přistupovat k obecným informacím z domovské stránky iSeries, která je nyní na následující webové stránce:
<http://www-1.ibm.com/servers/eserver/series/>

Přístup k seminářům týkajícím se rozšířených funkcí iSeries, použijte aplikaci Technical Studio na adrese: <http://www.iseries.ibm.com/tstudio/>

Ať sídlíte kdekoli, můžete si na internetu získat informace o opravách (PTF - program temporary fix) pro server iSeries, vybrat si je, objednat je a obdržet. Informace o PTF pro iSeries (ke stažení) a informace PSP (Preventive Service Planning) jsou k dispozici na této internetové adrese: <http://as400service.ibm.com>

iSeries Navigator

IBM iSeries Navigator je výkonné grafické rozhraní pro správu serverů iSeries. Funkce produktu iSeries Navigator zahrnují navigaci v systému, konfiguraci, funkce pro plánování a online nápovědu, jež vás provede vašimi úlohami. Produkt iSeries Navigator umožňuje snazší a výkonnější provoz a administraci serveru. Navíc je to jediné uživatelské rozhraní k novým rozšířeným funkcím operačního systému OS/400. Zahrnuje také funkci Centrální správa, pomocí které je možné spravovat několik serverů z centrálního systému.

Další informace o produktu iSeries Navigator najdete v aplikaci IBM iSeries Information Center a na následující webové stránce: <http://www.ibm.com/eserver/series/navigator/>.

Jak zasílat připomínky

Informace od vás jsou pro nás velmi důležité, abychom mohli vytvářet co nejpřesnější a velmi kvalitní technickou dokumentaci. Pokud máte nějaké připomínky k této publikaci nebo jiné dokumentaci k produktům OnDemand, laskavě nám je zašlete. K zaslání připomínek můžete zvolit některý z těchto způsobů:

- Ze Spojených států amerických, z Kanady a Portorika faxujte na číslo 1-800-937-3430 (z jiných zemí na číslo: 1-507-253-5192)
- Svě připomínky zasílejte elektronickou poštou na adresu: RCHCLERK@us.ibm.com; pokud jde o aplikaci Information Center, připomínky zasílejte na adresu: RCHINFO@us.ibm.com

Neopomeňte uvést následující informace:

- Název publikace v rámci aplikace iSeries Information Center.
- Číslo publikace (je uváděno v dolním pravém rohu titulní strany publikace).
- Číslo stránky nebo téma, kterého se vaše poznámka týká.

Přehled změn

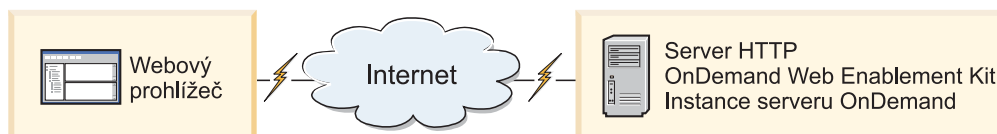
Toto vydání publikace *IBM Content Manager OnDemand for iSeries Common Server: Web Enablement Kit - Průvodce instalací a konfigurací* obsahuje nové a změněné technické informace. Mohou se sice vyskytovat některé příklady toho, kde byly provedeny změny, ale pruhy označující změny zde chybí. Mezi důležité změny, které je nutné vést v patrnosti, patří tyto skutečnosti:

- Ve verzi 5, vydání 1 zavádí produkt Content Manager OnDemand for iSeries (OnDemand) novou serverovou implementaci známou jako OnDemand Common Server. Common Server poskytuje vylepšené indexování, vyhledávání, prohlížení, zabezpečení, PDF a webové schopnosti pro uživatele a správce serveru OnDemand. Aktuální zákazníci OnDemand, kteří implementovali produkt Spool File Archive (s funkcí AnyStore nebo existující funkcí Server, nebo bez nich), mohou nyní migrovat na nový Common Server podle pokynů uvedených v dodatku A v publikaci Content Manager OnDemand for iSeries Common Server Planning and Installation Guide. Všimněte si, že v celém dokumentu se odkazy na migraci dat Spool File Archive vztahují také na data AnyStore, pokud je funkce AnyStore nainstalována.
- V publikaci Content Manager OnDemand for iSeries Common Server Indexing Reference byly provedeny významné změny týkající se funkcí podporovaných službou OS/400 Indexer. K těmto přidaným částem patří témata související s definováním indexů s více klíči, transakčních polí, textových vyhledávacích polí, souborů SCS pro souběžný tisk s překryvy AFP a masek pro aplikační pole.
- Produkt Content Manager OnDemand for iSeries nyní podporuje nové optické knihovny Plasmon podporované na serveru iSeries.
- Dva parametry pro příkaz STRASMOND byly odstraněny, aby bylo možné příkaz používat snadněji. Podrobnosti najdete v dodatku A v publikaci Content Manager OnDemand for iSeries Common Server Administration Guide.
- Systém OS/400 už nepodporuje originální podporu HTTP serveru. V této souvislosti byla také odstraněna podpora Content Manager OnDemand Web Enablement Kit (ODWEK) pro originální HTTP server. Server HTTP Apache je nyní jediný podporovaný HTTP server pro aplikaci ODWEK.

Kapitola 1. Přehled

Aplikace ODWEK umožňuje uživatelům přistupovat k datům, která jsou uložena na serveru IBM Content Manager OnDemand pomocí webového prohlížeče nebo uživatelsky napsaného programu. Například můžete některým lidem dát k dispozici adresu URL (Uniform Resource Locator) webové stránky, která jim umožní přihlásit se k serveru OnDemand; jiným lidem můžete dát k dispozici URL webové stránky, která jim umožní prohledat určitou složku. Aplikace ODWEK ověří, zda uživatelské informace jsou na serveru OnDemand platné, například povolení pro přístup k serveru a k datům uloženým ve skupině aplikací. Jakmile uživatel zadá vyhledávání, ODWEK zobrazí webovou stránku, která obsahuje seznam dokumentů, jež odpovídají dotazu. Uživatel vybere dokument, který se má zobrazit, a aplikace ODWEK dokument odešle do prohlížeče.

Obrázek 1 zobrazí pracovní stanici s webovým prohlížečem, který se používá pro přístup k datům ze serveru OnDemand.



Obrázek 1. Přístup k datům uloženým na serveru OnDemand pomocí ODWEK

Aplikace ODWEK může vyhledat a načíst dokumenty ze serverů OnDemand, které používají IBM Content Manager OnDemand pro iSeries Common Server, verze 5, IBM Content Manager OnDemand for Multiplatforms verze 7 a IBM Content Manager OnDemand pro z/OS a OS/390, verze 7.1.

Aplikace ODWEK obsahuje několik komponent:

- Programovací rozhraní OnDemand. Programovací rozhraní používá standardní rozhraní a protokoly OnDemand pro přístup k datům uloženým na serveru OnDemand. Na serveru OnDemand není zapotřebí žádný další kód pro podporu aplikace ODWEK. Pro řízení aplikace ODWEK můžete použít jedno z následujících programovacích rozhraní:
 - Program Common Gateway Interface (CGI). Program CGI poskytuje způsob pro přístup k datům OnDemand z webového prohlížeče. Program CGI se spouští v systému, který používá server HTTP (Hypertext Transfer Protocol), například IBM HTTP Server.
 - Java servlet. Program CGI poskytuje způsob pro přístup k datům OnDemand z webového prohlížeče. Tento servlet se spouští na HTTP serveru s povolenou Javou, který má spuštěný aplikační server Java, například IBM WebSphere Application Server.
 - Java API. Rozhraní Java API poskytuje způsob pro přístup k datům OnDemand z programu napsaného uživatelem. Rozhraní Java API vyžaduje Javu, verzi 1.2.2 nebo pozdější.
- IBM OnDemand Advanced Function Presentation (webový prohlížeč AFP). Webový prohlížeč AFP umožňuje uživatelům vyhledávat, načítat, zobrazit, procházet a tisknout dokumenty AFP z webového prohlížeče.
- Prohlížeč IBM OnDemand Image Web Viewer. Prohlížeč Image Web Viewer umožňuje uživatelům vyhledávat, načítat, zobrazovat, procházet a tisknout dokumenty BMP, GIF, JPEG, PCX a TIFF z webového prohlížeče.
- Java applet Line Data. Applet Line Data umožňuje uživatelům zobrazit dokumenty řádkových dat z webového prohlížeče. Administrátor umožňuje použití appletu Line Data konfiguračním souborem ARSWWW.INI.

- Java applet AFP2HTML. Applety AFP2HTML umožňují uživatelům zobrazit výstup generovaný nabízením služby IBM AFP2WEB Transform. Služba AFP2WEB Transform konvertuje dokumenty a prostředky AFP do souborů HTML (Hypertext Markup Language), které mohou být zobrazeny appletem AFP2HTML. Po instalaci a konfiguraci služby AFP2WEB Transform administrátor povolí používání appletu AFP2HTML konfigurací souboru ARSWWW.INI.

Poznámka: Chcete-li zobrazit jiné typy dokumentů uložených na serveru OnDemand, musíte získat a instalovat vhodný prohlížeč. Například chcete-li zobrazit dokumenty ve formátu PDF (Adobe Portable Data Format), IBM doporučuje, abyste získali prohlížeč Adobe Acrobat pro webové prohlížeče, které se používají ve vaší organizaci.

O programovacích rozhraních

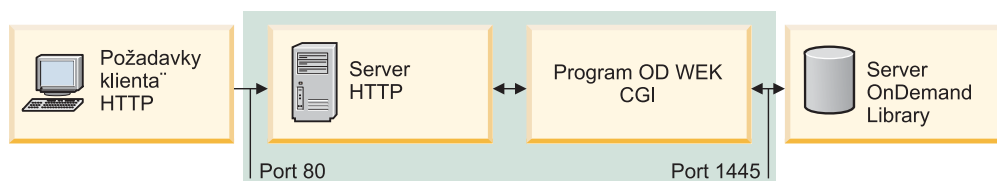
Instance aplikace ODWEK je kód ODWEK, který přistupuje k datům na serveru OnDemand. Instance řídí, co lze dělat s daty, a spravuje systémové prostředky, které jsou této instanci přiřazeny. Každá instance je samostatné prostředí. Instance má vlastní soubor ASWWW.INI a programovací rozhraní ODWEK, k němuž jiné instance nemohou přistupovat. Existují tři programovací rozhraní ODWEK:

- Program CGI, rozhraní mezi webovým prohlížečem a serverem OnDemand.
- Java servlet, rozhraní mezi webovým prohlížečem a serverem OnDemand.
- Rozhraní Java API, sada metod, kterou lze použít pro přístup k datům OnDemand z uživatelem napsaného programu

Je velmi důležité chápat, že instance může používat pouze programovací rozhraní. Programovací rozhraní se vzájemně vylučují. Není možné je použít v určité instanci zároveň. Je však možné spustit více instancí ODWEK na jednom počítači a přimět každou instanci používat jiné programovací rozhraní tak, že pro každou konfiguraci konfigurujete jiné číslo portu.

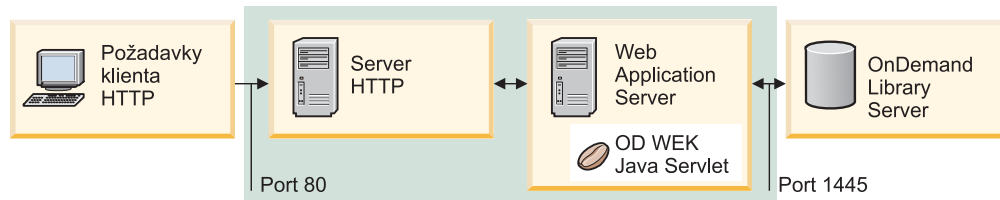
Nejobvyklejší implementací ODWEK je jediná instance v systému. Konfigurace s jedinou instancí je obvykle pro vývojáře nebo samostatné počítače a také pro instanci jednoho aplikačního serveru pracující nezávisle na jiných aplikacích.

Obrázek 2 uvádí příklad jedné instance používající rozhraní CGI.



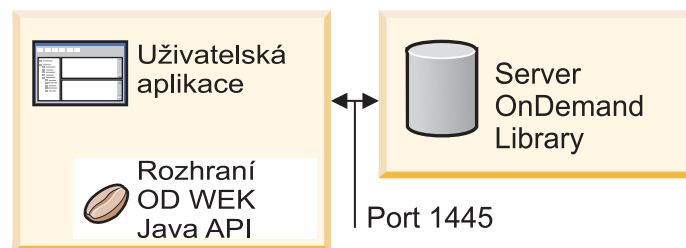
Obrázek 2. Jediná instance používající rozhraní CGI

Obrázek 3 na stránce 3 uvádí příklad jediné instance používající rozhraní Java servletu.



Obrázek 3. Jediná instance používající rozhraní Java

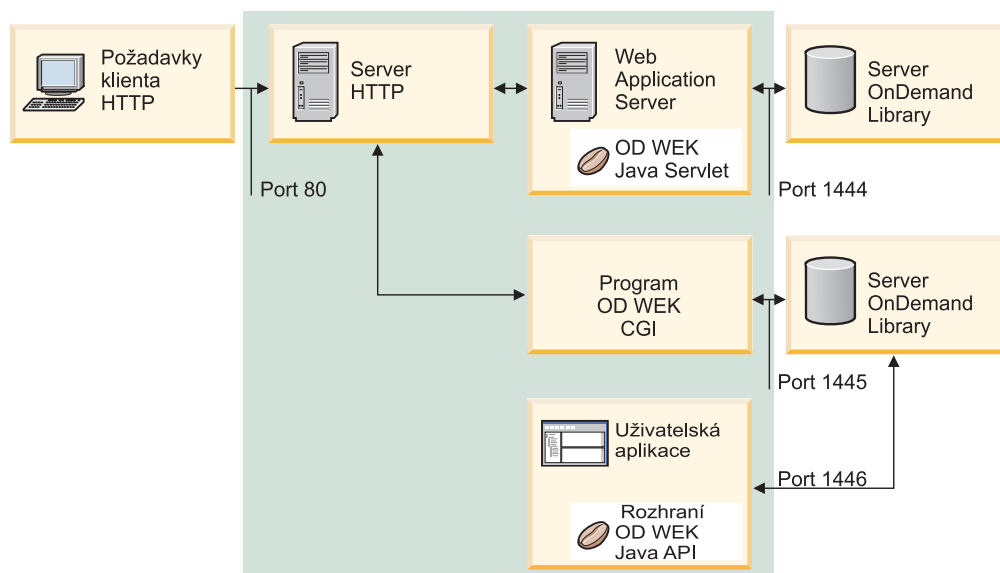
Obrázek 4 uvádí příklad jediné instance používající rozhraní Java API.



Obrázek 4. Jediná instance používající rozhraní Java API

Můžete konfigurovat více instancí ODWEK ve stejném systému. Každá instance vyžaduje své vlastní programovací rozhraní a soubor ARSWWW.INI, který uvádí jediné číslo portu, na kterém se odehrává komunikace mezi programovacím rozhraním a serverem OnDemand. Každá instance také vyžaduje své vlastní uložení a zabezpečení. Konfigurace s více instancemi je obvykle pro zákazníky, kteří potřebují spouštět jednu nebo více vývojářských, testovacích nebo výrobních aplikací ve stejném systému. Instance pracují nezávisle na sobě.

Obrázek 5 uvádí příklad topologie s více instancemi.



Obrázek 5. Topologie s více instancemi

O prohlížečích

Aplikace ODWEK obsahuje následující prohlížeče:

- AFP Web Viewer.
- Image Web Viewer.
- Java applet Line Data.
- Java applet AFP2HTML.

Prohlížeč AFP Web Viewer a Image Viewer jsou softwarové programy, které určitým způsobem rozšiřují schopnosti webového prohlížeče. Webový prohlížeč umožňuje uživatelům AFP prohlížet dokumenty AFP. Prohlížeč obrázků umožňuje uživatelům prohlížet dokumenty ve formátu BMP, GIF, JPEG, PCX a TIFF. Prohlížeče umožňují zobrazovat dokumenty v okně prohlížeče. Každý prohlížeč přidává panel nástrojů do horní části zobrazovacího okna. Panel nástrojů prohlížeče může být doplňkem k panelu nástrojů prohlížeče. Přidaný panel nástrojů obsahuje ovládací prvky, které pomáhají uživatelům pracovat s dokumenty. Pracovníci ve vaší organizaci, kteří chtějí prohlížet dokumenty pomocí webových prohlížečů, musí webové prohlížeče instalovat na svých pracovních stanicích.

Poznámka: Instalační program nainstaluje prohlížeče buď jako programy typu plugin, nebo jako ovládací prvky ActiveX. Pokud je na pracovní stanici instalován Internet Explorer, pak instalační program nainstaluje ovládací prvky ActiveX; pokud je na pracovní stanici nainstalován Netscape, pak instalační program nainstaluje programy typu plugin. Pokud na pracovní stanici máte instalovaný Internet Explorer a Netscape, instalační program nainstaluje ovládací prvky ActiveX pro Internet Explorer a programy typu plugin pro Netscape.

| Applet Line Data umožňuje uživatelům prohlížet SCS a napojené datové dokumenty, které
| jsou uloženy na serveru OnDemand. Applet Line Data zobrazuje napojené datové dokumenty
| v okně prohlížeče a přidává panel nástrojů k horní části okna prohlížeče. Panel nástrojů
| appletu Line Data poskytuje ovládací prvky, které mohou uživatelům pomoci pracovat
| s dokumenty. Administrátor umožňuje použití appletu Line Data konfiguračním souborem
| ARSWWW.INI.

Applety AFP2HTML umožňují uživatelům zobrazit výstup generovaný nabízením služby IBM AFP2WEB Transform. Služba AFP2WEB Transform konvertuje dokumenty a prostředky AFP na dokumenty HTML. Po instalaci a konfiguraci služby AFP2WEB Transform administrátor povolí používání appletu AFP2HTML konfiguračním souborem ARSWWW.INI. Applet AFP2HTML poskytuje panel nástrojů s ovládacími prvky, které mohou uživatelům pomoci pracovat s dokumenty, včetně ovládacích prvků pro velké objekty.

Jednou výhodou appletů je to, že vaši uživatelé nemusí nikdy instalovat nebo inovovat software na pracovní stanici, aby je mohli používat, na rozdíl od webových prohlížečů, které musí být instalovány na pracovní stanici. Navíc, když IBM poskytne novou verzi webového prohlížeče, musíte aktualizovaný webový prohlížeč distribuovat svým uživatelům.

Když používáte applety a prohlížeče od IBM, dokumenty, které jsou načteny ze serveru OnDemand, zůstanou komprimovány, dokud nedosáhnou klienta. Klient dokumenty zobrazí a zobrazí je v okně webového prohlížeče. Jestliže byl dokument uložen v OnDemand jako velký objekt, pak klient načte a rozbálí segmenty dokumentu podle potřeby, jak uživatel prochází stránkami dokumentu.

Použití ODWEK

Nejběžnější metodou používání aplikace ODWEK je přizpůsobení vzorových aplikací HTML, které jsou dodány s produktem. Vzorová aplikace LOGON.HTM podporuje uživatele, kteří mají povolen přístup k několika složkám. Nejprve upravte stránku LOGON.HTM s informacemi o svém serveru OnDemand. Pak můžete publikovat URL ze souboru LOGON.HTM. Vaši uživatelé se pak mohou připojit k URL a přihlásit se k zadanému serveru. Aplikace ODWEK automaticky zobrazí řadu webových stránek, které mohou uživatelé prohledávat, načítat a zobrazovat dokumenty OnDemand. Vzorové aplikace CREDIT.HTM podporují nahodilé použití OnDemand tím, že nabízejí webovou stránku, která obsahuje vyhledávací kritéria pro specifickou složku. Jakmile si vzor upravíte, uživatel se připojí k URL, dokončí vyhledávací kritéria a stiskne tlačítko Submit. Aplikace ODWEK zobrazí webovou stránku, která vypisuje dokumenty, které odpovídají dotazu.

Důležité: Aplikace ODWEK vyžaduje schopnost pro zápis dat objektů cookie na klientovi. Zajistěte, aby vaši uživatelé konfigurovali své prohlížeče pro příjem objektů cookie.

Většina zákazníků definuje jeden ID uživatele OnDemand pro k přístup k serverům s ODWEK. To je běžné v prostředích s mnoha příležitostnými uživateli OnDemand, kteří budou přistupovat ke stejné složce. Každému uživateli můžete také přidělit vlastní ID uživatele OnDemand. Nehledě na to, jak se rozhodnete přistupovat k aplikaci ODWEK, musíte spravovat ID uživatele v OnDemand: musíte je přidat do serveru a nastavit pro uživatele skupinu aplikací a oprávnění ke složkám.

Funkce produktu

Aplikace ODWEK podporuje následující funkce OnDemand. Funkce se obvykle vyvolávají vytvořením webových stránek, které obsahují odkazy na serverový program ODWEK. Každým odkazem se vyvolává specifická funkce. Výstupem jedné funkce je jiná webová stránka s odkazy, které vedou uživatele k dalším logickým funkcím. Například počáteční webová stránka, může vyvolávat funkce k přihlášení. Funkce k přihlášení generuje webovou stránku s odkazem na funkci Search Criteria (Vyhledávací kritéria). Každá funkce může být volána z rozhraní API (Application Programming Interface). Podrobnosti viz téma Dodatek A, "Referenční informace pro rozhraní CGI API", na stránce 53.

Funkce Add Annotation (Přidání anotace)

Funkce Add Annotation (Přidání anotace) umožňuje uživatelům přidávat anotaci k zadanému dokumentu. K přidávání anotace musí uživatel mít oprávnění pro přidávání anotace pro každou skupinu aplikací, která obsahuje dokumenty, jež mají být anotovány. (Oprávnění Application Group Access umožňuje uživatelům přidávat anotace.)

Funkce Change Password (Změna hesla)

Funkce Change Password (Změna hesla) umožňuje uživatelům měnit jejich hesla na serveru OnDemand.

Funkce Document Hit List (Seznam nalezených dokumentů)

Funkce Document Hit List (Seznam nalezených dokumentů) vytváří seznam položek, která odpovídají vyhledávacím kritériím. Seznam je uveden v tabulce HTML. Každá položka, která odpovídá kritériím vyhledávání, se uloží do pole v tabulce a obsahuje odkaz na funkci Retrieve Document (Načtení dokumentu).

Důležitá poznámka pro zákazníky, kteří mají v systému obě prostředí OnDemand Spool File Archive a Common Server a používají záznam ARS_MIGR_SERVER v souboru

ARS.CFG ke kombinaci složek Spool File Archive a Common Server v jediném seznamu pro výběr složek: Pro uživatele ODWEK se v seznamu složek ODWEK zobrazí složky Spool File Archive a bude možné je prohledávat. Avšak pokus o načtení dokumentu selže.

Funkce Logoff (Odhlášení)

Funkce Logoff (Odhlášení) umožňuje uživatelům odhlášení od serveru OnDemand.

Funkce Logon (Přihlášení)

Funkce Logon (Přihlášení) umožňuje uživatelům přihlášení k serveru OnDemand. Jestliže funkce Logon (Přihlášení) je úspěšná, uživatel obdrží webovou stránku, která obsahuje seznam složek, které je uživatel oprávněn otevřít.

Funkce Retrieve Document (Načtení dokumentu)

Funkce Retrieve Document (Načtení dokumentu) načítá dokument ze serveru OnDemand. Datový tok vrácený ze serveru zahrnuje dokument a v závislosti na typu dat i prostředky požadované k zobrazení dokumentu. Datový tok nesmí být žádným způsobem pozměněn. Prohledávací program spolu s prohlížečem mohou interpretovat a dekódovat datový tok a zobrazit dokument. Jestliže je dokument uložen na serveru OnDemand jako velký objekt, pak se vrátí pouze první segment dokumentu. Následující segmenty dokumentu lze podle potřeby načíst a zobrazit.

Funkce Search Criteria (Vyhledávací kritéria)

Po úspěšném přihlášení obdrží uživatel seznam složek, které je uživatel oprávněn otevřít. Uživatel vybere složku, kterou chce otevřít. Po otevření složky se zobrazí webová stránka, která obsahuje vyhledávací pole pro složku. Uživatel může přijmout předvolená vyhledávací kritéria nebo zadat vyhledávací kritéria pro vyhledání specifických dokumentů. Když uživatel stiskne tlačítko Submit (Odeslat), požadavek na vyhledání se odešle do serveru OnDemand.

Funkce Server Print Document (Tisk dokumentů na serveru)

Funkce Server Print Document (Tisk dokumentů na serveru) odesílá kopie dokumentů na tiskárnu serveru OnDemand. Pro tisk na serveru musí uživatel mít oprávnění Document Print ke každé skupině aplikací, která obsahuje dokumenty, které uživatel potřebuje vytisknout. (Oprávnění Application Group Access umožňuje uživatelům tisknout dokumenty.) Na serveru OnDemand musí být definována alespoň jedna serverová tiskárna.

Funkce Update Document (Aktualizace dokumentu)

Funkce Update Document (Aktualizace dokumentu) umožňuje uživatelům aktualizovat databázi. Funkce Update Document (Aktualizace dokumentu) aktualizuje jedno nebo více databázových polí pro určitý dokument.

Funkce View Annotations (Zobrazení anotací)

Funkce View Annotations (Zobrazení anotací) umožňuje uživatelům zobrazit anotace připojené k zadanému dokumentu. K zobrazení anotací musí mít uživatel oprávnění Annotation View ke každé skupině aplikací, která obsahuje anotace, jež si uživatel potřebuje prohlédnout. (Oprávnění Application Group Access umožňuje uživatelům prohlížet anotace.)

Zabezpečení serveru a dat

Existují dvě úrovně zabezpečení, které je nutné zvážit před použitím aplikace ODWEK:

- Kdo může přistupovat k programům ODWEK a webovým stránkám.
- Kdo může přistupovat k datům na serveru OnDemand.

Každý uživatel může přistoupit k vašemu HTTP serveru a k programům a webovým stránkám, které představují "front-end", aby aplikace ODWEK mohla potenciálně přistupovat k datům uloženým na serveru OnDemand. IBM důrazně doporučuje, abyste omezili přístup k programům a webovým stránkám. Existuje mnoho způsobů, jak můžete omezit přístup k programům a webovým stránkám na HTTP serveru. Například na mnoha HTTP serverech se zabezpečení citlivých webových stránek provádí tím, že omezíte přístup k adresářům. Můžete také použít soubor hesel na HTTP serveru, který od uživatelů před přístupem na webové stránky vyžaduje zadání ID uživatele a hesla. Třebaže se však ID uživatelů a hesla na HTTP serveru podobají ID uživatelů a heslům v operačním systému, není mezi nimi žádná souvislost. Není také žádná souvislost mezi ID uživatelů a hesly na HTTP serveru a mezi ID uživatelů a hesly na serveru OnDemand.

Aplikace ODWEK umožňuje přístup k serverům OnDemand a datům prostřednictvím standardních rozhraní API OnDemand. Rozhraní API ověřují, zda ID uživatelů OnDemand mohou přistupovat k serveru a požadovaným datům. Někdo ve vaší organizaci musí spravovat zabezpečení uživatelů a dat na serveru OnDemand.

Existuje ještě jeden faktor související se zabezpečením, který musíte zvážit: metoda použitá pro přenos formulářových parametrů a hodnot mezi klientem a serverem. Formuláře opatřené aplikací ODWEK používají pro přenos parametrů a hodnot v těle požadavku HTTP metodu POST. Při použití metody POST se parametry a hodnoty nezobrazují v poli Location (Umístění) prohledávacího programu. Například typické volání funkce vypadá následovně:

```
http://www.company.com/cgi-bin/arswww.cgi
```

Jestliže však nezadáte metody, když vytváříte formulář, pak předvolená metoda je GET - přenáší parametry a hodnoty v rámci samotného URL. Typické volání funkce metodou GET vypadá následovně:

```
http://www.company.com/cgi-bin/arswww.cgi?_function=logon  
&_user=bob&_password=secret
```

Parametry a hodnoty se objevují jako text v poli Location (Umístění) v okně prohledávacího programu. Jestliže vytváříte své vlastní formuláře, IBM důrazně doporučuje, abyste používali metodu POST. Chcete-li změnit předvolenou metodu z GET na POST, musíte zahrnout atribut METHOD do kódu na příznaku form.

Poznámka: Jestliže musíte používat metodu GET, pak můžete parametry a hodnoty šifrovat zadáním parametru ENCRYPTURL do souboru ARSWWW.INI. Další informace najdete v tématu "ENCRYPTURL" na stránce 32.

Kapitola 2. Instalace a konfigurace HTTP serveru

Tato část uvádí požadavky pro instalaci a vysvětluje, jak instalovat software ODWEK na HTTP server, a jak upravit konfigurační soubor ODWEK.

Nejprve je třeba nainstalovat software ODWEK do systému iSeries, v němž je aktuální verze serveru IBM HTTP Server. Pokud navíc plánujete používat Java servlet, pak ověřte, že máte spuštěnou aktuální verzi webového aplikačního serveru iSeries (WebSphere).

Aplikace ODWEK může prohledávat a načítat dokumenty ze serverů OnDemand, které používají IBM Content Manager OnDemand pro iSeries verze 5, vydání 3, Common Server.

Požadavky na instalaci

ODWEK vyžaduje:

- Aktuální verzi produktu IBM HTTP Apache Server. Pokud navíc plánujete používat Java servlet, pak ověřte, že máte spuštěnou aktuální verzi produktu iSeries Web Application Server (WebSphere). Servery musí být spuštěny na OS/400, verze 5, vydání 3.
- Vhodný typ média pro instalaci.
- uostatečný diskový prostor pro instalační soubory: přibližně 30 MB na HTTP server.
- Dostatečný diskový prostor pro rychlou vyrovnávací paměť: je předvoleno 10 MB na HTTP server. Další informace najdete v tématu "CACHE SIZE" na stránce 15.

Jiné požadavky

Aplikace ODWEK může ukládat dokumenty do rychlé vyrovnávací paměti (cache) na HTTP serveru. Tím lze urychlit poskytování dříve zobrazovaných dokumentů uživatelům. Chcete-li povolit ukládání do rychlé vyrovnávací paměti, konfiguruje parametr CACHEDOCS do souboru ARSWWW.INI. Podrobnosti najdete v tématu "CACHEDOCS" na stránce 14.

V rámci předvolby ukládá aplikace ODWEK data do adresáře /QIBM/UserData/OnDemand/WWW/CACHE. Můžete zadat jiný adresář rychlé vyrovnávací paměti tím, že upravíte soubor ARSWWW.INI. Podrobnosti viz téma "CACHEDIR" na stránce 14.

Zajistěte, aby procesy, které používají programy ODWEK, mohly číst z adresáře, který obsahuje programy, a mohly zapisovat do adresáře rychlé vyrovnávací paměti (cache). Když je instalována aplikace ODWEK, všechny objekty jsou zabezpečeny seznamem oprávnění QONDADM a uživatelské profily QTMHHTTP, QTMHHTTP1 a QEJBSVR budou přidány do seznamu oprávnění s oprávněním *CHANGE. V seznamu oprávnění QRDARS400 musí být také uvedeny uživatelské profily QTMHHTTP, QTMHHTTP1 a QEJBSVR s oprávněním *USE.

Jestliže plánujete použít applet AFP2HTML, musíte si zakoupit službu AFP2WEB od IBM, nainstalovat ji a nakonfigurovat ji na HTTP serveru. Další informace o nabízené službě AFP2WEB Transform vám sdělí odpovědný pracovník IBM. Musíte také uvést konfigurační volby pro dokumenty a prostředky AFP, které plánujete zpracovat s aplikací AFP2WEB Transform. Další informace o konfiguračním souboru najdete v tématu Dodatek E, "Konverze AFP na HTML", na stránce 123.

Jestliže plánujete konvertovat dokumenty AFP uložené na serveru OnDemand na soubory PDF, které lze prohlížet programem Adobe Acrobat, musíte získat službu AFP2PDF

Transform od IBM, instalovat ji a konfigurovat ji na HTTP serveru. Další informace o nabízené službě AFP2WEB Transform vám sdělí odpovědný pracovník IBM. Musíte také uvést konfigurační volby pro dokumenty a prostředky AFP, které plánujete zpracovat s aplikací AFP2PDF Transform. Další informace o konfiguračním souboru najdete v tématu Dodatek F, “Konverze AFP na PDF”, na stránce 127. Chcete-li zobrazit konvertované dokumenty, musíte si obstarat prohlížeč Adobe Acrobat pro prohlížeče, které používáte ve vaší organizaci.

Instalace v OS/400

Nastavení aplikace ODWEK obvykle vyžaduje, abyste provedli následující:

1. Získejte kopii posledního souboru OnDemand README. Než začnete, vytiskněte a pak přečtete celý soubor.
2. Chcete-li instalovat aplikaci ODWEK, postupujte podle pokynů v knize nazvané *Instalace softwaru* (SC41-5120). Číslo licencovaného programu je 5722RD1 a funkce je 11.

Poznámka: Doporučovaným způsobem, jak nainstalovat aplikaci ODWEK, je provést instalaci pomocí menu Instalovat licencované programy z menu Práce s licencovanými programy (go licpgm). Na obrazovku Install licensed programs zadejte 1 jako volbu Add, zadejte 5722RD1 jako Licensed Program a 11 pro položku Product Option nebo procházejte seznamem Licensed Programs and Product Options, až naleznete položku ODWEK a zadejte před ní 1. **Pokud nainstalujete produkt OnDemand jinak, než bylo uvedeno výše, mohou se během používání tohoto produktu vyskytnout chyby.**

3. IBM doporučuje, abyste si objednali, nahráli a aplikovali všechna dostupná PTF pro produkt OnDemand. Tyto činnosti proveďte po úspěšné instalaci licencovaného programu. Informační APAR II13680 obsahuje úplný seznam PTF pro produkt OnDemand verze 5, vydání 3. Informační APAR si můžete objednat elektronicky pomocí příkazu SNDPTFORD, ve kterém uvedete jako číslo PTF hodnotu II13680. Ujistěte se, že jste si přečetli průvodní dopisy k tomuto PTF a postupujte podle všech speciálních instrukcí.

Váš další krok

Ověřte, že máte spuštěnou aktuální verzi serveru IBM HTTP Server v systému iSeries. Budete muset nakonfigurovat HTTP server. Příklad souboru pro konfiguraci serveru najdete v tématu Dodatek G, “Konfigurační soubory HTTP serveru”, na stránce 129.

Pokud plánujete používat Java servlet, pak ověřte, že máte spuštěnou aktuální verzi webového aplikačního serveru iSeries (WebSphere). Budete muset konfigurovat server WebSphere. Pokyny najdete v tématu IBM WebSphere Application Server for AS/400 Documentation Center na adrese www.ibm.com/servers/eserver/iseries/software/websphere/wsappserver/. Sledujte odkazy pro instalaci a počáteční konfiguraci pro příslušnou verzi serveru WebSphere.

Jakmile nainstalujete software ODWEK, nakonfigurujete HTTP server a (případně) nakonfigurujete WebSphere, můžete nakonfigurovat inicializační soubor ODWEK pro prostředí svého operačního systému. Viz téma “Zadání souboru ARSWWW.INI”.

Zadání souboru ARSWWW.INI

Soubor ARSWWW.INI je textový soubor ASCII obsahující parametry, které jsou čteny programy ODWEK (například program CGI nebo Java servlet). Každý parametr můžete zadat na zvláštním řádku v následujícím formátu: **PARAMETER=hodnota**. Příklad:

```
AFPVIEWING=plugin
CACHEDIR=/tmp/cache
LANGUAGE=ENU
```

Parametry v souboru ARSWWW.INI jsou uspořádány do sekcí. Začátek sekce můžete zadat pomocí hlavičky sekce, a to v následujícím formátu: [hlavičkaSekce]. Parametry pro sekci můžete zadat za hlavičku sekce. Příklad:

```
[@SRV@_gunnar]
HOST=gunnar
PORT=1446
PROTOCOL=0
```

S produktem se dodává vzorový konfigurační soubor ARSWWW.INI. Příklad konfiguračního souboru obsahuje nejčastěji používané hodnoty. Téma “Příklad souboru ARSWWW.INI” na stránce 36 uvádí příklad.

Toto jsou sekce a parametry pro soubor ARSWWW.INI:

[@SRV@_DEFAULT]

Předvolená sekce serveru. Předvolenou sekci serveru můžete použít k zadání parametrů, které jsou běžné pro servery OnDemand, s nimiž aplikace ODWEK komunikuje. Použijí se parametry a hodnoty, které zadáte do této sekce, pokud ale nemáte zadané parametry do serverové sekce.

Tato sekce má globální rozsah pro všechny servery a do souboru ARSWWW.INI ji můžete zadat pouze jednou.

Tato sekce je volitelná.

Tato sekce může obsahovat následující parametry:

PORT

Číslo portu TCP/IP, který servery OnDemand používají pro komunikaci s aplikací ODWEK. Jestliže neuvedete parametr PORT, pak server použije číslo portu, které je zadáno pro server OnDemand v tabulce služeb (WRKSRVTBLE). Jestliže nezádáte hodnotu pro parametr PORT a server OnDemand není uveden v tabulce služeb, servery budou pokoušet použít číslo portu 1445. Chcete-li zajistit, aby servery používaly číslo portu, které je zadáno pro server OnDemand v tabulce služeb, zadejte 0 (nula).

Tento parametr budete muset zadat jednou do předvolené sekce. Když používáte rozhraní Logon API, můžete zadané číslo portu potlačit parametrem `_port`.

Tento parametr je volitelný.

Příklad:

```
[@SRV@_DEFAULT]
PORT=0
```

PROTOCOL

Síťový protokol, který servery OnDemand používají ke komunikaci s aplikací ODWEK. Pro TCP/IP musíte zadat 0 (nula).

Tento parametr musíte do předvolené sekce zadat pouze jednou.

Tento parametr je volitelný. Jestliže tento parametr nezádáte, použije se hodnota 0 (nula).

Příklad:

```
[@SRV@_DEFAULT]  
PROTOCOL=0
```

[@SRV@_server]

Serverová sekce. Musíte zadat jednu serverovou sekci pro každý server OnDemand, s nímž aplikace ODWEK bude komunikovat. Serverová sekce obsahuje parametry a hodnoty pro zadaný server. Hlavička sekce musí obsahovat řetězec, který server identifikuje. Parametry a hodnoty zadané v serverové sekci potlačí parametry nacházející se v předvolené serverové sekci.

Pro každý server musíte zadat jednu serverovou sekci.

Tato sekce se požaduje.

Tato sekce může obsahovat následující parametry:

HOST

Jméno serveru OnDemand. Můžete zadat adresu TCP/IP, alias jména hostitele nebo plně kvalifikované jméno hostitele serveru.

Tento parametr musíte do serverové sekce zadat pouze jednou.

Tento parametr je povinný.

Příklad:

```
[@SRV@_gunnar]  
HOST=gunnar
```

PORT

Číslo portu TCP/IP, které server OnDemand používá ke komunikaci s aplikací ODWEK. Jestliže neuvedete parametr PORT, pak server použije číslo portu, které je zadáno (nebo předvoleno) v předvolené serverové sekci.

Tento parametr můžete do serverové sekce zadat pouze jednou. Když používáte rozhraní Logon API, můžete zadané číslo portu potlačit parametrem `_port`.

Tento parametr je volitelný.

Příklad:

```
[@SRV@_gunnar]  
PORT=0
```

Toto číslo portu by mělo odpovídat číslu portu zadanému v souboru `ars.ini` pro danou instanci.

PROTOCOL

Síťový protokol, který server OnDemand používá pro komunikaci s aplikací ODWEK. Pro TCP/IP musíte zadat 0 (nula).

Tento parametr můžete do serverové sekce zadat pouze jednou.

Tento parametr je volitelný. Pokud není zadán, použije se hodnota zadaná (nebo předvolená) v předvoleném serveru.

Příklad:

```
[@SRV@_gunnar]  
PROTOCOL=0
```

[CONFIGURATION]

Sekce CONFIGURATION obsahuje parametry, které se používají v aplikaci ODWEK na HTTP serveru.

Tato sekce má globální rozsah a do souboru ARSWWW.INI ji můžete zadat pouze jednou.

Tato sekce je volitelná.

Tato sekce může obsahovat následující parametry:

APPLETCACHEDIR

Uvádí adresář, do kterého applet Line Data a applet AFP2HTML dočasně ukládají dokumenty. Adresář může být lokální na pracovní stanici uživatele nebo na síťové jednotce. Všichni uživatelé musí mít k zadanému adresáři přístup pro zápis.

Příklad:

```
[Configuration]  
APPLETCACHEDIR=/QIBM/UserData/OnDemand/www/cache
```

Poznámky:

1. Parametr APPLETCACHEDIR má globální rozsah.
2. Parametr APPLETCACHEDIR je volitelný. Jestliže však tento parametr není zadán, applety se budou pokoušet dokumenty ukládat do pracovního adresáře Java.
3. Jestliže zadaný adresář neexistuje, applety se pokusí dokumenty ukládat do pracovního adresáře Java.
4. Applet odstraní dokument z adresáře paměti cache, když uživatel opustí applet (například zavře dokument).

APPLETDIR

Označuje adresář, který obsahuje palety Line Data a AFP2HTML.

Poznámky:

1. Můžete zadat jméno adresáře nebo AliasMatch:
 - Jestliže zadáte jméno adresáře, adresář se musí vztahovat k adresáři /QIBM/UserData/OnDemand/WWW. Například když zadáte appletdir=applets, applety musí existovat v adresáři /QIBM/UserData/OnDemand/WWW/APPLETS.
 - Jestliže zadáte AliasMatch, musí být v konfiguračním souboru definován HTTP server. Když například zadáte appletdir=/applets/, konfigurační soubor HTTP serveru musí mít AliasMatch pro /applets/. Cesta souboru výměny v pravidlu AliasMatch musí být nastavena na jméno úplné cesty k adresáři na serveru. Příklad:

```
AliasMatch ^/applets/com/ibm/edmslod/(.*)$ /QIBM/UserData/OnDemand/www/applets/$1  
AliasMatch ^/applets/(.*)$ /QIBM/UserData/OnDemand/www/applets/$1
```
2. Ověřte oprávnění k adresáři, který zadáte. Procesy, které spouští programy ODWEK, musí číst adresář appletů.

Tento parametr má globální rozsah a můžete jej v sekci CONFIGURATION zadat pouze jednou.

Tento parametr je povinný.

Příklad:

```
[CONFIGURATION]
APPLETDIR=applets
```

CACHEDIR

Používá se k zadání adresáře na HTTP serveru, na který aplikace ODWEK ukládá (*do rychlé vyrovnávací paměti*) dokumenty (viz téma “CACHEDOCS”). Je předvoleno, že aplikace ODWEK ukládá dokumenty do adresáře /QIBM/UserData/OnDemand/WWW/CACHE.

Poznámka: Ověřte oprávnění k adresáři, který zadáte. Procesy, které spouštějí programy ODWEK, musí zapisovat do adresáře rychlé vyrovnávací paměti (cache) a číst z tohoto adresáře.

Tento parametr má globální rozsah a můžete jej v sekci CONFIGURATION zadat pouze jednou.

Tento parametr je volitelný.

Příklad:

```
[CONFIGURATION]
CACHEDIR=/QIBM/UserData/OnDemand/WWW/CACHE
```

CACHEDOCS

Určuje, zda ODWEK dočasně ukládá (*do rychlé vyrovnávací paměti*) dokumenty na HTTP serveru. Ukládáním do rychlé vyrovnávací paměti lze urychlit načítání dříve zobrazených dokumentů ze serveru. Předvolená hodnota je 0 (nula), což znamená, že ukládání dokumentů do rychlé vyrovnávací paměti není povoleno. Chcete-li povolit ukládání do rychlé vyrovnávací paměti, zadejte 1 (jedna). Jestliže povolíte ukládání dokumentů do rychlé vyrovnávací paměti, ověřte adresář, do kterého aplikace ODWEK ukládá dokumenty (viz téma “CACHEDIR”), a velikost prostoru na disku vyhrazeného pro rychlou vyrovnávací paměť (viz téma “CACHESIZE” na stránce 15).

Poznámka: IBM doporučuje, abyste pro dokumenty vždy povolili ukládání do rychlé vyrovnávací paměti, když používáte prohlížeč Microsoft Internet Explorer a webový prohlížeč AFP nebo Image Web Viewer.

Tento parametr má globální rozsah a můžete jej v sekci CONFIGURATION zadat pouze jednou.

Tento parametr je volitelný. Obecně však platí, že většina zákazníků by měla povolit ukládání dokumentů do rychlé vyrovnávací paměti.

Příklad:

```
[CONFIGURATION]
CACHEDOCS=1
```

CACHEMAXTHRESHOLD

Určuje, kdy aplikace ODWEK začne mazat data a dokumenty z rychlé vyrovnávací paměti. Aplikace ODWEK začne mazat data a dokumenty, když se diskový prostor využitý rychlou vyrovnávací pamětí rovná zadané hodnotě nebo je vyšší. Předvolená hodnota je 80 (osmdesát procent). Aplikace ODWEK maže nejstarší položky z rychlé vyrovnávací paměti, dokud nebude práh překročen (viz téma “CACHEMINTHRESHOLD” na stránce 15).

Tento parametr má globální rozsah a můžete jej v sekci CONFIGURATION zadat pouze jednou.

Tento parametr je volitelný.

Příklad:

```
[CONFIGURATION]  
CACHEMAXTHRESHOLD=80
```

CACHEMINTHRESHOLD

Určuje, kdy aplikace ODWEK přestane mazat data a dokumenty z rychlé vyrovnávací paměti. Aplikace ODWEK přestane mazat data a dokumenty, když se diskový prostor využitý rychlou vyrovnávací pamětí rovná zadané hodnotě nebo je nižší. Předvolená hodnota je 40 (čtyřicet procent). Aplikace ODWEK začne mazat nejstarší položky z rychlé vyrovnávací paměti, jakmile bude práh dosažen (viz téma “CACHEMAXTHRESHOLD” na stránce 14).

Tento parametr má globální rozsah a můžete jej v sekci CONFIGURATION zadat pouze jednou.

Tento parametr je volitelný.

Příklad:

```
[CONFIGURATION]  
CACHEMINTHRESHOLD=40
```

CACHESIZE

Velikost diskového prostoru, který může aplikace ODWEK používat pro dočasné ukládání (*rychlá vyrovnávací paměť*) dat a dokumentů na HTTP serveru. Zadejte hodnotu v megabajtech. Předvolená hodnota je 10 (deset megabajtů).

Poznámka: Chcete-li povolit ukládání dokumentů do rychlé vyrovnávací paměti, viz téma “CACHEDOCS” na stránce 14.

Tento parametr má globální rozsah a můžete jej v sekci CONFIGURATION zadat pouze jednou.

Tento parametr je volitelný. Když ukládáte dokumenty do rychlé vyrovnávací paměti, čím více diskového prostoru přiřadíte, tím více dokumentů ODWEK je možné uložit na HTTP serveru. Obecně to může zvýšit rychlost, jakou aplikace ODWEK odesílá dříve zobrazené dokumenty uživatelům.

Příklad:

```
[CONFIGURATION]  
CACHESIZE=1024
```

CACHEUSERIDS

Uvádí seznam oddělený čárkami, obsahující ID uživatelů OnDemand. Aplikace ODWEK pro něj používá data z rychlé vyrovnávací paměti k dokončení procesu přihlášení. Pro zadaná ID uživatelů bude provedeno více pokusů o přihlášení, které přeskočí standardní proces přihlášení k serveru OnDemand, kromě případů, kdy data nejsou v rychlé vyrovnávací paměti nebo kdy je dosažena hodnota Inactivity Time Out (viz systémové parametry na serveru OnDemand). Každý ID uživatelů oddělte čárkou.

Poznámky:

1. Jestliže se v ID uživatele na serveru rozlišují malá a velká písmena (viz systémové parametry na serveru OnDemand), pak musíte zadat ID uživatele přesně tak, jak byl definovaný na serveru OnDemand.

2. ID uživatele uvedený v seznamu CACHEUSERIDS může přistupovat pouze k těm složkám, jejichž jména a jiné informace jsou v rychlé vyrovnávací paměti. Uživatelé nebudou schopni přistoupit ke složkám vytvořeným po jejich přihlášení k serveru OnDemand. Chcete-li povolit ID uživatele uvedeného v seznamu CACHEUSERIDS přístup k nějaké nové složce, buď vymažte jméno uživatele ze seznamu CACHEUSERIDS, nebo vyčistěte rychlou vyrovnávací paměť.
3. Chcete-li zadat, že aplikace ODWEK by měla používat data z rychlé vyrovnávací paměti pro všechny uživatele OnDemand, zadejte `CACHEUSERIDS=*`.

Tento parametr má globální rozsah a můžete jej v sekci CONFIGURATION zadat pouze jednou.

Tento parametr je volitelný.

Příklad:

```
[CONFIGURATION]
CACHEUSERIDS=user1,user2,user3
```

CODEPAGE

Označuje kódovou stránku databáze OnDemand. Je předvoleno, že aplikace ODWEK používá kódovou stránku HTTP serveru.

Tento parametr má globální rozsah a můžete jej v sekci CONFIGURATION zadat pouze jednou. Když používáte rozhraní Logon API, můžete potlačit zadanou kódovou stránku parametrem `_codepage`.

Tento parametr je volitelný. Jestliže však HTTP server používá jinou kódovou stránku než databáze, musíte zadat parametr CODEPAGE.

Příklad:

```
[CONFIGURATION]
CODEPAGE=37
```

DOCSIZE

Při načítání dokumentů ze serveru OnDemand tento parametr určuje maximální velikost (v bajtech) dokumentu, který lze zapsat přímo do paměti namísto prvního zápisu dokumentu na disk. Každý dokument, který je menší než nebo rovný uvedené hodnotě, se запиše přímo do paměti. Každý dokument, který přesahuje zadanou hodnotu, se nejprve запиše na disk, načte se do paměti a pak se předá do prohlížeče. Nižší hodnota šetří systémové prostředky, kdežto vyšší hodnota zdokonalí výkon při prohlížení. Rozsah je od 0 (nula) do n bajtů, kde n je paměť dostupná v systému. Při zadané nulové hodnotě se použije předvolená hodnota 1 MB. Pokud tento parametr není uveden nebo pokud hodnota není definována nebo rozpoznána, použije se předvolená velikost 1 MB.

Tento parametr má globální rozsah a můžete jej v sekci CONFIGURATION zadat pouze jednou.

Tento parametr je volitelný.

Příklad:

```
[CONFIGURATION]
DOCSIZE=524287
```

IMAGEDIR

Označuje adresář, který obsahuje soubory obrázků používané aplikací ODWEK.

Poznámky:

1. Aplikace ODWEK spojí vámi zadanou hodnotu a jména souborů zadaná v HTML příznacích obrázku. Když například uvedete:

```
imagedir=pictures
```

Pak příznak obrázku HTML pro funkci View Document se ve výstupu zobrazí následovně:

```
<IMG SRC="pictures/odic_vd.gif">
```

2. Můžete zadat jméno adresáře nebo AliasMatch:

- Jestliže zadáte jméno adresáře, pak se adresář musí vztahovat k adresáři /QIBM/UserData/OnDemand/WWW. Když například zadáte imagedir=pictures, pak obrázky musí existovat v adresáři /QIBM/UserData/OnDemand/WWW/PICTURES.

- Jestliže zadáte pravidlo AliasMatch, pak musí být definováno v konfiguračním souboru HTTP serveru. Když například zadáte imagedir=/pictures/, pak v konfiguračním souboru HTTP serveru musí být uvedeno AliasMatch pro /pictures/. Pravidlo AliasMatch musí být nastaveno na jméno úplné cesty adresáře na serveru. Příklad:

```
AliasMatch ^/images/(.*)$ /QIBM/UserData/OnDemand/WWW/PICTURES/$1
```

3. Ověřte oprávnění k adresáři, který zadáte. Procesy, které spouští programy ODWEK, musí číst adresář obrázků.

Tento parametr má globální rozsah a můžete jej v sekci CONFIGURATION zadat pouze jednou.

Tento parametr je povinný.

Příklad:

```
[CONFIGURATION]  
IMAGEDIR=pictures
```

LANGUAGE

Označuje jazyk, ve kterém aplikace ODWEK zobrazuje zprávy. Předvolný jazyk je angličtina (ENU). Aplikace ODWEK podporuje následující jazyky:

Hodnota	Území
ARA	Egypt
CHS	Čína
CHT	Taiwan
DAN	Dánsko
DEU	Německo
ENU	USA / angličtina
ESP	Španělsko
FIN	Finsko
FRA	Francie
FRC	Kanada
ITA	Itálie
JPN	Japonsko
KOR	Korea
NLD	Nizozemí

Hodnota	Území
NOR	Norsko
PTB	Brazílie
SVE	Švédsko

Tento parametr má globální rozsah a můžete jej v sekci CONFIGURATION zadat pouze jednou.

Tento parametr je volitelný.

Příklad:

```
[CONFIGURATION]
LANGUAGE=JPN
```

ShowSearchString

Určuje, zda je funkce Auto Find aktivní nebo neaktivní. Funkce Auto Find podporuje transakce a vyhledávání textu v dokumentech typu "line data" z Java prohlížeče Line Data Viewer. Funkce Auto Find automaticky vyhledá a zvýrazní určitý řádek v libovolném dokumentu, který odpovídá vyhledávacím kritériím, jež zadal uživatel.

Když je funkce Auto Find aktivována a uživatel provede transakci nebo vyhledání textu a otevře dokument z výsledného seznamu dokumentů, systém automaticky vyhledá text dokumentu zadaný ve vyhledávacím kritériu. Po nalezení textu odpovídajícího vyhledávacím kritériím se zvýrazní řádek obsahující příslušný text; pokud nebude text nalezen, zobrazí se odpovídající zpráva. Když uživatel otevře jiný dokument pro prohlédnutí (nebo znovu otevře již dříve prohlížený dokument), vyhledání se provede znovu.

Chcete-li aktivovat funkci Auto Find, nastavte parametr ShowSearchString na hodnotu 1 (jedna). Chcete-li deaktivovat funkci Auto Find, nastavte parametr ShowSearchString na hodnotu 0 (nula).

Tento parametr má globální rozsah a můžete jej v sekci CONFIGURATION zadat pouze jednou.

Tento parametr je volitelný. Jestliže není uveden, předvolená hodnota je 0 (nula; neaktivní).

Příklad:

```
[CONFIGURATION]
ShowSearchString=1
```

TEMPDIR

Tento parametr používejte k zadání adresáře, do kterého aplikace ODWEK bude ukládat dočasné soubory.

Tento parametr má globální rozsah a můžete jej v sekci CONFIGURATION zadat pouze jednou.

Tento parametr je volitelný. Jestliže neuvedete parametr TEMPDIR, aplikace ODWEK bude ukládat dočasné soubory do runtime adresáře. Jestliže používáte program CGI, runtime adresář je adresář, do kterého byl instalován program CGI. Jestliže používáte servlet, runtime adresář je adresář, který obsahuje servlet: pro některé instalace je runtime adresář ten adresář, kde je umístěn soubor java.exe file; pro jiné je runtime adresář ten, kde jsou servlety, avšak přesné umístění závisí na aplikačním serveru Java.

Příklad:

```
[CONFIGURATION]
TEMPDIR=/QIBM/UserData/OnDemand/WWW/TMP
```

Poznámka: Ověřte oprávnění k adresáři, který zadáte. Procesy, které spouštějí programy ODWEK musí zapisovat do dočasného adresáře a číst z něho.

TEMPLATEDIR

Označuje adresář, který obsahuje soubory šablon HTML. Aplikace ODWEK používá soubory šablon ke generování webových stran v reakci na různé funkce produktu (například přihlášení, vyhledávání, načtení dokumentu atd.). Je předvoleno, že aplikace ODWEK načítá soubory šablon z adresáře /QIBM/UserData/OnDemand/WWW/SAMPLES.

Poznámka: Ověřte oprávnění k adresáři, který zadáte. Procesy, které spouští programy ODWEK, musí číst adresář šablon.

Tento parametr má globální rozsah a můžete jej v sekci CONFIGURATION zadat pouze jednou.

Tento parametr je volitelný.

Příklad:

```
[CONFIGURATION]
TEMPLATEDIR=/QIBM/UserData/OnDemand/WWW/SAMPLES
```

[SECURITY]

Sekce SECURITY obsahuje zabezpečovací parametry, které se používají v aplikaci ODWEK na HTTP serveru.

Tato sekce má globální rozsah a do souboru ARSWWW.INI ji můžete zadat pouze jednou.

Tato sekce je volitelná.

Tato sekce může obsahovat následující parametry:

REPORTSERVERTIMEOUT

Používá se k zadání toho, že aplikace ODWEK by měla používat parametr Inactivity Time Out ze serveru OnDemand. Parametr Inactivity Time Out určuje, kdy server může ukončit relaci s neaktivním uživatelem. Chcete-li zadat, že aplikace ODWEK má používat parametr Inactivity Time Out, nastavte parametr REPORTSERVERTIMEOUT na hodnotu 1 (jedna).

Tento parametr má globální rozsah a můžete jej v sekci SECURITY zadat pouze jednou.

Tento parametr je volitelný. Jestliže neuvedete parametr REPORTSERVERTIMEOUT, pak aplikace ODWEK nebude používat parametr Inactivity Time Out, což znamená, že aplikace ODWEK neukončí relaci s neaktivním uživatelem. Další informace o parametru Inactivity Time Out najdete v online nápovědě pro administrativního klienta.

Příklad:

```
[SECURITY]
REPORTSERVERTIMEOUT=1
```

SERVERACCESS

Uvádí seznam dělený čárkami, který obsahuje servery OnDemand, k nimž může aplikace ODWEK přistupovat. Jestliže zadáte parametr SERVERACCESS, klienti, kteří používají

aplikaci ODWEK, a programy, které používají rozhraní API, mají povolen přístup pouze k těm serverům, které zadáte. Můžete zadat adresu TCP/IP, alias jména hostitele nebo plně kvalifikované jméno hostitele serveru.

Tento parametr má globální rozsah a můžete jej v sekci SECURITY zadat pouze jednou.

Tento parametr je volitelný.

Příklad:

```
[SECURITY]  
SERVERACCESS=dave,gunnar
```

[AFP2HTML]

Sekce AFP2HTML obsahuje parametry, které používá služba AFP2WEB Transform. Služba AFP2WEB Transform konvertuje dokumenty a prostředky AFP do souborů HTML, které mohou být zobrazeny appletem AFP2HTML.

Poznámky:

1. Aby bylo možné konvertovat dokumenty AFP na dokumenty HTML, administrátor musí získat službu AFP2WEB Transform od IBM, instalovat ji a konfigurovat na serveru. Další informace o nabízené službě AFP2WEB Transform vám sdělí odpovědný pracovník IBM. Někdo ve vaší organizaci musí také nastavit konfigurační volby pro službu AFP2WEB Transform. Další informace o konfiguračním souboru najdete v tématu Dodatek E, “Konverze AFP na HTML”, na stránce 123.
2. Aby bylo možné konvertovat dokumenty službou AFP2WEB Transform, musíte zadat parametr AFPVIEWING=HTML do sekce DEFAULT BROWSER (nebo do sekcí jiných prohlížečů). Podrobnosti viz téma “AFPVIEWING” na stránce 30. (Jestliže plánujete používat rozhraní Retrieve Document API, musíte zadat parametr _afp=HTML. Podrobnosti najdete v tématu “Funkce Retrieve Document (Načtení dokumentu)” na stránce 71.)
3. Je předvoleno, že aplikace ODWEK používá k zobrazení konvertovaných dokumentů applet AFP2HTML. Jestliže konvertovaný dokument byl uložen do serveru OnDemand jako velký objekt, pak applet AFP2HTML zajistí řídicí prvky, aby uživatelům pomohl snadno přecházet na libovolnou stranu dokumentu.

Tato sekce má globální rozsah a do souboru ARSWWW.INI ji můžete zadat pouze jednou.

Tato sekce je volitelná.

Tato sekce může obsahovat následující parametry:

CONFIGFILE

Konfigurační soubor, který obsahuje volby použité službou AFP2WEB Transform ke konverzi dokumentů a prostředků AFP na data, fonty a obrázky HTML, které lze prohlížet appletem AFP2HTML. Dodatek E, “Konverze AFP na HTML”, na stránce 123 uvádí vzorový konfigurační soubor, který se dodává k serveru OnDemand. V dokumentaci ke službě AFP2WEB Transform naleznete podrobnosti o volbách, které můžete zadat v konfiguračním souboru.

Tento parametr má globální rozsah a můžete jej v sekci AFP2HTML zadat pouze jednou.

Tento parametr je volitelný.

Příklad:

```
[AFP2HTML]
CONFIGFILE=afp2html.ini
```

INSTALLDIR

Adresář, který obsahuje programy AFP2WEB Transform, konfigurační soubory a mapovací soubory. Zadejte jméno úplné cesty k adresáři na HTTP serveru.

Poznámka: Ověřte oprávnění k adresáři, který zadáte. Procesy, které spouští programy ODWEK, musí číst instalační adresář.

Tento parametr má globální rozsah a můžete jej v sekci AFP2HTML zadat pouze jednou.

Tento parametr je volitelný.

Příklad:

```
[AFP2HTML]
INSTALLDIR=/QIBM/UserData/OnDemand/www/bin
```

USEEXECUTABLE

Určuje, zda aplikace ODWEK spouští službu AFP2WEB Transform pomocí sdílené knihovny (DLL) nebo spustitelného souboru (EXE).

Důležité: Aplikace ODWEK na serveru iSeries musí používat spustitelný soubor. Proto musí být tento parametr vždy nastaven na hodnotu 1 (jedna).

Tento parametr má globální rozsah a můžete jej v sekci AFP2HTML zadat pouze jednou.

Tento parametr je volitelný.

Příklad:

```
[AFP2HTML]
USEEXECUTABLE=1
```

[AFP2PDF]

Sekce AFP2PDF obsahuje parametry, které používá služba IBM AFP2PDF Transform. Služba AFP2PDF Transform konvertuje dokumenty a prostředky AFP na PDF dokumenty, které lze prohlížet programem Adobe Acrobat.

Poznámky:

1. Aby bylo možné konvertovat dokumenty AFP na PDF dokumenty, administrátor musí získat službu AFP2PDF Transform od IBM, instalovat ji a konfigurovat na HTTP serveru. Další informace o nabízené službě AFP2WEB Transform vám sdělí odpovědný pracovník IBM. Někdo ve vaší organizaci musí také nastavit konfigurační volby pro službu AFP2PDF Transform. Další informace o konfiguračním souboru najdete v tématu Dodatek F, “Konverze AFP na PDF”, na stránce 127.
2. Chcete-li konvertovat dokumenty službou AFP2PDF Transform, musíte uvést parametr `AFPVIEWING=PDF` v sekci `DEFAULT BROWSER` (nebo v sekcích jiného prohlížeče). Podrobnosti viz téma “AFPVIEWING” na stránce 30. (Jestliže plánujete používat rozhraní Retrieve Document API, musíte zadat parametr `_afp=PDF`. Podrobnosti najdete v tématu “Funkce Retrieve Document (Načtení dokumentu)” na stránce 71.)
3. Je předvoleno, že aplikace ODWEK používá k zobrazení konvertovaných dokumentů prohlížeč Adobe Acrobat. Tento prohlížeč musíte získat pro prohlížeče, které se používají ve vaší organizaci.

Tato sekce má globální rozsah a do souboru ARSWWW.INI ji můžete zadat pouze jednou.

Tato sekce je volitelná.

Tato sekce může obsahovat následující parametry:

CONFIGFILE

Konfigurační soubor, který obsahuje volby používané službou AFP2PDF Transform ke konverzi dokumentů a prostředků AFP na dokumenty PDF, které lze prohlížet programem Adobe Acrobat. Dodatek F, “Konverze AFP na PDF”, na stránce 127 uvádí vzorový konfigurační soubor dodávaný se serverem OnDemand. V dokumentaci ke službě AFP2PDF Transform naleznete podrobnosti o volbách, které můžete zadat v konfiguračním souboru.

Tento parametr má globální rozsah a můžete jej v sekci AFP2PDF zadat pouze jednou.

Tento parametr je volitelný.

Příklad:

```
[AFP2PDF]
CONFIGFILE=afp2pdf.ini
```

INSTALLDIR

Adresář, který obsahuje programy AFP2PDF Transform, konfigurační soubory a mapovací soubory. Zadejte jméno úplné cesty k adresáři na HTTP serveru.

Poznámka: Ověřte oprávnění k adresáři, který zadáte. Procesy, které spouští programy ODWEK, musí číst instalační adresář.

Tento parametr má globální rozsah a můžete jej v sekci AFP2PDF zadat pouze jednou.

Tento parametr je volitelný.

Příklad:

```
[AFP2PDF]
INSTALLDIR=/QIBM/UserData/OnDemand/www/bin
```

USEEXECUTABLE

Určuje, zda aplikace ODWEK spouští službu AFP2WEB Transform pomocí sdílené knihovny (DLL) nebo spustitelného souboru (EXE).

Důležité: Aplikace ODWEK na serveru iSeries musí používat spustitelný soubor. Proto musí být tento parametr vždy nastaven na hodnotu 1 (jedna).

Tento parametr má globální rozsah a můžete jej v sekci AFP2PDF zadat pouze jednou.

Tento parametr je volitelný.

Příklad:

```
[AFP2PDF]
USEEXECUTABLE=1
```

[MIMETYPES]

Sekce MIMETYPES označuje typ obsahu Multipurpose Internet Mail Extension (MIME) pro dokumenty, které budou načteny ze serveru OnDemand. Prohlížeč používá typ obsahu MIME pro formátování a zobrazení dokumentu, pro výběr správného appletu nebo prohlížeče k otevření dokumentu, nebo pro spuštění uživatelsky definovaného programu k otevření dokumentu.

Poznámky:

1. Sekce MIMETYPES by měla obsahovat dvojici **parameter=hodnota** pro každý typ dokumentu, který plánujete načítat ze serveru OnDemand. Hodnota **parameter** označuje typ dat v dokumentu na serveru OnDemand. (Tento typ dat je přiřazen k aplikaci OnDemand na straně View Information.) Parametr *hodnota* určuje program, který se spouští k otevření dokumentu. V parametru *value* se rozlišují malá a velká písmena.
2. V příkladu souboru ARSWWW.INI (viz téma “Příklad souboru ARSWWW.INI” na stránce 36) obsahuje sekce MIMETYPES parametr pro všechny standardní datové typy podporované na serveru OnDemand (AFP, BMP, EMAIL, GIF, JFIF, LINE, PCX, PDF a TIFF).
3. Kromě standardních datových typů server OnDemand také podporuje uživatelsky definované datové typy. Uživatelsky definovaný datový typ může označovat libovolný jiný dat dat, která chcete ukládat v systému. Než budou uživatelé moci prohlížet dokumenty, které mají uživatelsky definovaný datový typ, musíte přidat parametr do sekce MIMETYPE. Parametr musí uvádět typ datového obsahu MIME a příponu souboru, která byla zadána pro aplikaci OnDemand na straně View Information. Přípona souboru musí být také registrována v operačním systému klienta. Předpokládáme například, že definujete aplikaci pro ukládání dokumentů Lotus WordPro na serveru OnDemand. Na stránce View Information v aplikaci zadáte příponu souboru jako LWP. Chcete-li konfigurovat systém tak, aby rozpoznával dokumenty načtené z aplikace, přidejte následující parametr do souboru ARSWWW.INI:

```
[MIMETYPES]
LWP=application/vnd.lotus-wordpro
```

Když pak uživatel načítá dokument z aplikace, ODWEK nastaví typ obsahu MIME na `application/vnd.lotus-wordpro` a systém spustí Lotus WordPro pro otevření dokumentu. Pro Netscape musí být typ obsahu MIME definován v rámci volby Preferences -> Navigator -> Applications.

Tabulka 1 uvádí typy obsahu MIME pro několik PC aplikací:

Tabulka 1. Typy obsahu MIME pro několik PC aplikací

Aplikace	Typy obsahu MIME
Aplikace Lotus	WK1=application/vnd.lotus-1-2-3 WK3=application/vnd.lotus-1-2-3 WK4=application/vnd.lotus-1-2-3 123=application/vnd.lotus-1-2-3 APR=application/vnd.lotus-approach VEW=application/vnd.lotus-approach LWP=application/vnd.lotus-wordpro SAM=application/vnd.lotus-wordpro MWP=application/vnd.lotus-wordpro SMM=application/vnd.lotus-wordpro PRE=application/vnd.lotus-freelance PRZ=application/vnd.lotus-freelance

Tabulka 1. Typy obsahu MIME pro několik PC aplikací (pokračování)

Aplikace	Typy obsahu MIME
Aplikace Microsoft	DOC=application/msword XLS=application/vnd.ms-excel PPS=application/vnd.ms-powerpoint PPT=application/vnd.ms-powerpoint MPD=application/vnd.ms-project MPP=application/vnd.ms-project MPT=application/vnd.ms-project MPD=application/vnd.ms-project
Aplikace HTML	HTML=application/html HTM=application/htm

Tato sekce má globální rozsah a do souboru ARSWWW.INI ji můžete zadat pouze jednou.

Tato sekce je volitelná.

Tato sekce může obsahovat následující parametry:

AFP

Typ obsahu MIME pro dokumenty AFP, když AFPVIEWING=NATIVE je uveden v sekci [DEFAULT BROWSER]. Další informace najdete v tématu “AFPVIEWING” na stránce 30. Uvádí typ MIME pro dokument, který prohlížeč pak používá pro určení toho, jaký program by se měl použít ke zpracování dokumentu.

Tento parametr má globální rozsah a můžete jej v sekci MIMETYPES zadat pouze jednou.

Tento parametr je volitelný.

Příklad:

```
[MIMETYPES]
AFP=application/afp
```

BMP

Typ obsahu MIME pro dokumenty BMP. Je předvoleno, že dokumenty BMP se zobrazují webovým prohlížečem obrázků.

Tento parametr má globální rozsah a můžete jej v sekci MIMETYPES zadat pouze jednou.

Tento parametr je volitelný. Jestliže však tento parametr neuvedete, pak aplikace ODWEK nastaví typ obsahu MIME na image/bmp a spustí program, který je přiřazen k typu souboru BMP v operačním systému klienta.

Příklad:

```
[MIMETYPES]
BMP=image/IBM-OnDemand
```

GIF

Typ obsahu MIME pro dokumenty GIF. Je předvoleno, že dokumenty GIF se zobrazují webovým prohlížečem obrázků.

Tento parametr má globální rozsah a můžete jej v sekci MIMETYPES zadat pouze jednou.

Tento parametr je volitelný. Jestliže však neuvédete tento parametr, pak ODWEK nastaví typ obsahu MIME na `image/gif` a použije k zobrazení souborů GIF prohlížeč implementovaný do webového prohlížeče.

Příklad:

```
[MIMETYPES]
GIF=image/IBM-OnDemand
```

EMAIL

Typ obsahu MIME pro dokumenty EMAIL. Informace o zpracování dokumentů EMAIL před jejich odesláním klientovi najdete v tématu “EMAILVIEWING” na stránce 31.

Poznámky:

1. Jestliže provedete konverzi dokumentů EMAIL na HTML, aplikace ODWEK nastaví typ obsahu MIME na `text/html`. Aplikace ODWEK ignoruje hodnotu parametru EMAIL, pokud je uvedena.
2. Jestliže rozbalíte a dekomprimujete dokumenty EMAIL ze serveru OnDemand, aplikace ODWEK použije hodnotu parametru EMAIL pro určení programu k otevření dokumentu.

Tento parametr má globální rozsah a můžete jej v sekci MIMETYPES zadat pouze jednou.

Tento parametr je volitelný.

Příklad:

```
[MIMETYPES]
EMAIL=text/plain
```

JFIF

Typ obsahu MIME pro dokumenty JFIF (JPEG). Je předvoleno, že dokumenty JFIF se zobrazují webovým prohlížečem obrázků.

Tento parametr má globální rozsah a můžete jej v sekci MIMETYPES zadat pouze jednou.

Tento parametr je volitelný. Jestliže však tento parametr neuvédete, pak aplikace ODWEK nastaví typ obsahu MIME na `image/jpeg` a spustí program, který je přiřazen k typu souboru JPEG v operačním systému klienta.

Příklad:

```
[MIMETYPES]
JFIF=image/IBM-OnDemand
```

LINE

Typ obsahu MIME pro dokumenty Line Data. Informace o zpracování dokumentů Line Data před jejich odesláním klientovi najdete v tématu “LINEVIEWING” na stránce 32.

Používá se, když je v sekci [DEFAULT BROWSER] zadáno `LINEVIEWING=NATIVE`. Jestliže rozbalíte a dekomprimujete dokumenty Line Data ze serveru OnDemand, aplikace ODWEK použije hodnotu parametru LINE pro určení programu k otevření dokumentu.

Tento parametr má globální rozsah a můžete jej v sekci MIMETYPES zadat pouze jednou.

Tento parametr je volitelný.

Příklad:

```
[MIMETYPES]
LINE=text/html
```

PCX

Typ obsahu MIME pro dokumenty PCX. Je předvoleno, že dokumenty PCX se zobrazují webovým prohlížečem obrázků.

Tento parametr má globální rozsah a můžete jej v sekci MIMETYPES zadat pouze jednou.

Tento parametr je volitelný. Jestliže však tento parametr neuvédete, pak aplikace ODWEK nastaví typ obsahu MIME na `image/pcx` a spustí program, který je přiřazen k typu souboru PCX v operačním systému klienta.

Příklad:

```
[MIMETYPES]
PCX=image/IBM-OnDemand
```

PDF

Typ obsahu MIME pro dokumenty PDF.

Poznámky:

1. Aplikace ODWEK použije hodnotu parametru PDF pro určení programu k otevření PDF dokumentů. Je předvoleno, že dokumenty PDF se otvírají prohlížečem Adobe Acrobat.
2. Chcete-li zobrazit dokumenty PDF, měli byste získat a instalovat prohlížeč Adobe Acrobat pro webové prohlížeče, které se používají ve vaší organizaci.

Tento parametr má globální rozsah a můžete jej v sekci MIMETYPES zadat pouze jednou.

Tento parametr je volitelný.

Příklad:

```
[MIMETYPES]
PDF=application/pdf
```

TIFF

Typ obsahu MIME pro dokumenty TIFF. Je předvoleno, že dokumenty TIFF se zobrazují webovým prohlížečem obrázků.

Tento parametr má globální rozsah a můžete jej v sekci MIMETYPES zadat pouze jednou.

Tento parametr je volitelný. Jestliže však tento parametr neuvédete, pak aplikace ODWEK nastaví typ obsahu MIME na `image/tiff` a spustí program, který je přiřazen k typu souboru TIFF v operačním systému klienta.

Příklad:

```
[MIMETYPES]
TIFF=image/IBM-OnDemand
```

[ATTACHMENT IMAGES]

Sekce ATTACHMENT IMAGES označuje obrázkové soubory, které aplikace ODWEK používá k zobrazování příloh k dokumentu. Každý obrázkový soubor by měl obsahovat ikonu, která představuje určitý typ přípony. Můžete například zadat soubor obrázku, který obsahuje ikonu pro textovou přílohu, bitmapovou přílohu atd.

Poznámky:

1. Každý vámi zadaný parametr musí označovat typ souboru, který operační systém přiřadí k typu přílohy. Typ souboru určuje program, který operační systém spouští pro zpracování přílohy. Jestliže operační systém například přiřadí typ souboru TXT k textovým souborům, které jsou připojeny jako přílohy, přidejte parametr `TXT=hodnota`

do sekce ATTACHMENT IMAGES. Jako parametr *hodnota* zadejte jméno souboru, jež obsahuje ikonu, kterou chcete použít pro označení textové přílohy k dokumentu. Když uživatel klikne na ikonu, operační systém spustí program, který je registrován pro otevírání dokumentů TXT.

2. Je předvoleno, že všechny přílohy k dokumentu, jsou označeny souborem `odic_att.gif` (který je umístěn v adresáři, jež je zadán parametrem `IMAGEDIR` v sekci `CONFIGURATION`). Server OnDemand také používá soubor `odic_att.gif` pro každý z typů souborů, pro které není v sekci ATTACHMENT IMAGES uveden parametr.

Tato sekce má globální rozsah a do souboru ARSWWW.INI ji můžete zadat pouze jednou.

Tato sekce je volitelná.

Tato sekce může obsahovat následující parametry:

BMP

Parametr označuje bitmapový typ dat. Hodnota označuje soubor, který obsahuje ukonku představující bitmapový obrázek připojeny k dokumentu.

Tento parametr má globální rozsah a můžete jej v sekci ATTACHMENT IMAGES zadat pouze jednou.

Tento parametr je volitelný.

Příklad:

```
[ATTACHMENT IMAGES]
BMP=userBitMap.gif
```

GIF

Parametr označuje typ dat GIF. Hodnota označuje soubor, který obsahuje ukonku představující obrázek GIF připojeny k dokumentu.

Tento parametr má globální rozsah a můžete jej v sekci ATTACHMENT IMAGES zadat pouze jednou.

Tento parametr je volitelný.

Příklad:

```
[ATTACHMENT IMAGES]
GIF=userGIF.gif
```

TXT

Parametr označuje typ dat TXT. Hodnota označuje soubor, který obsahuje ikonu představující textový soubor připojeny k dokumentu.

Tento parametr má globální rozsah a můžete jej v sekci ATTACHMENT IMAGES zadat pouze jednou.

Tento parametr je volitelný.

Příklad:

```
[ATTACHMENT IMAGES]
TXT=userText.gif
```

[NO HTML]

Sekce NO HTML obsahuje parametry, které se používají pro potlačení předvolených znaků, které oddělují řetězce a seznamy hodnot v oddělovaném výstupu ASCII. Funkce generuje oddělovaný výstup ASCII, když její parametr `_nohtml` nastavíte na 1 (jedna). Podrobnosti o oddělovaném výstupu ASCII najdete v tématu Dodatek H, “Žádný HTML výstup”, na stránce 131.

Tato sekce má globální rozsah a do souboru ARSWWW.INI ji můžete zadat pouze jednou.

Tato sekce je volitelná.

Tato sekce může obsahovat následující parametry:

BEGIN

Znak, který aplikace ODWEK používá pro oddělení začátku řetězce nebo řetězce hodnot. Oddělovací znak BEGIN musíte změnit, jestliže řetězec obsahuje předvolený znak (znak []).

Tento parametr má globální rozsah a můžete jej v sekci NO HTML zadat pouze jednou.

Tento parametr je volitelný.

Příklad:

```
[NO HTML]
BEGIN=<
```

END

Znak, který aplikace ODWEK používá pro oddělení konce řetězce nebo řetězce hodnot. Oddělovací znak END musíte změnit, jestliže řetězec obsahuje předvolený znak (znak []).

Tento parametr má globální rozsah a můžete jej v sekci NO HTML zadat pouze jednou.

Tento parametr je volitelný.

Příklad:

```
[NO HTML]
END=>
```

SEPARATOR

Znak, který aplikace ODWEK používá pro oddělení hodnot v řetězci. Oddělovací znak SEPARATOR musíte změnit, jestliže řetězec obsahuje předvolený znak (znak ^).

Tento parametr má globální rozsah a můžete jej v sekci NO HTML zadat pouze jednou.

Tento parametr je volitelný.

Příklad:

```
[NO HTML]
SEPARATOR=;
```

[DEFAULT BROWSER]

Sekci DEFAULT BROWSER můžete použít k zadání parametrů pro prohlížeče, které se používají ve vaší organizaci. Vámi zadané parametry se budou používat, dokud je neuvedete v sekci specifického prohlížeče, jak je podrobně popsáno v tématu “[browser]” na stránce 35. (Parametry zadané v sekci prohlížeče potlačí ty, které jsou v sekci DEFAULT BROWSER.)

Tato sekce má globální rozsah pro všechny webové prohlížeče a do souboru ARSWWW.INI ji můžete zadat pouze jednou.

Tato sekce je volitelná.

Tato sekce může obsahovat následující parametry:

ADDEXTENSION

Určuje, zda tříznaková přípona souboru dokumentu se přidává k dodatečným informacím o cestě URL, které se vrací do prohlížeče. Přidání přípony souboru do URL může pomoci prohlížečům určit správný prohlížeč pro dokument. Předvolená hodnota je 0 (nula) a znamená, že přípona souboru se nepřidává do URL.

Poznámka: Jestliže používáte Microsoft Internet Explorer, IBM doporučuje, abyste uvedli ADDEXTENSION=1, aby se přípona souboru při dala do URL.

Tento parametr má globální rozsah a můžete jej v sekci DEFAULT BROWSER zadat pouze jednou.

Tento parametr je volitelný.

Příklad:

```
[DEFAULT BROWSER]
ADDEXTENSION=1
```

ADDFIELDSTODOCID

Určuje, zda se hodnoty v polích přidávají do identifikátorů dokumentů. (Identifikátory dokumentů jsou vráceny funkcí Document Hit List.) Předvolená hodnota je 0 (nula) a znamená, že hodnoty v polích se nepřidávají do identifikátorů dokumentů. Jestliže povolíte aplikaci ODWEK přidávat hodnoty v polích do identifikátorů dokumentů, objeví se také v systémovém protokolu za předpokladu, že jste konfigurovali systém pro ukládání zpráv skupiny aplikací do systémového protokolu.

Poznámky:

1. Jestliže použijete funkci Update Document, pak musíte uvést ADDFIELDSTODOCID=1.
2. Jestliže volba Annotation Flags in the document database table field je nastavena na Yes, pak **musíte** uvést ADDFIELDSTODOCID=1. Volbu Annotations Flags in document database table field v dialogovém okně Database Information můžete nastavit ze strany General v definicích skupiny aplikací OnDemand. (Klepnutím na položku Advanced otevřete dialogové okno Database Information.)

Tento parametr má globální rozsah a můžete jej v sekci DEFAULT BROWSER zadat pouze jednou.

Tento parametr je volitelný.

Příklad:

```
[DEFAULT BROWSER]
ADDFIELDSTODOCID=1
```

ADDNOTES

Určuje, zda lze k dokumentům přidávat anotace. Pokud je to povoleno, aplikace ODWEK umístí ovládací prvek pro přidávání anotací vedle každého dokumentu v seznamu dokumentů. Předvolená hodnota je 0 (nula) a znamená, že do dokumentů nelze přidávat anotace.

Poznámka: Uživatelé mohou nebo nemohou přidávat anotace do dokumentů, což závisí na oprávněních Annotation ve skupině aplikací OnDemand.

Tento parametr má globální rozsah a můžete jej v sekci DEFAULT BROWSER zadat pouze jednou.

Tento parametr je volitelný.

Příklad:

```
[DEFAULT BROWSER]
ADDNOTES=1
```

AFPVIEWING

Když uživatel načítá dokument AFP ze serveru OnDemand, hodnota tohoto parametru určuje, jakou akci - pokud vůbec nějakou - aplikace ODWEK provede, než dokument odešle klientovi. Někteří zákazníci například konvertují dokumenty AFP na dokumenty HTML pomocí služby AFP2WEB Transform a používají applet AFP2HTML k zobrazení HTML výstupu. Tito zákazníci by měli uvést AFPVIEWING=HTML, aby aplikace ODWEK konvertovala dokument AFP před tím, než je odešle klientovi.

Parametr můžete nastavit na jednu z následujících hodnot:

- | | |
|---------------|---|
| ASCII | Aplikace ODWEK konvertuje dokumenty AFP na ASCII text. |
| HTML | Aplikace ODWEK konvertuje dokumenty AFP na dokumenty HTML pomocí služby AFP2WEB Transform. |
| NATIVE | Aplikace ODWEK rozbaluje a dekomprimuje dokumenty AFP a jejich prostředky ze serveru OnDemand.

Poznámka: Jestliže zadáte AFPVIEWING=NATIVE, ověřte, že typ obsahu MIME pro dokumenty AFP označuje prohlížeč, který chcete používat. Podrobnosti viz téma “[MIMETYPES]” na stránce 22. |
| PDF | Aplikace ODWEK konvertuje dokumenty AFP na dokumenty PDF pomocí služby AFP2WEB Transform.

Poznámka: Jestliže zadáte AFPVIEWING=PDF, ověřte, že typ obsahu MIME pro dokumenty PDF označuje prohlížeč, který chcete používat. Podrobnosti viz téma “[MIMETYPES]” na stránce 22. |
| PLUGIN | Aplikace ODWEK nekonvertuje dokumenty AFP (předvolba). |

Tento parametr má globální rozsah a můžete jej v sekci DEFAULT BROWSER zadat pouze jednou. Když používáte funkci Retrieve Document, můžete zadanou akci potlačit parametrem `_afp`.

Tento parametr je volitelný.

Příklad:

```
[DEFAULT BROWSER]
AFPVIEWING=PLUGIN
```

AUTODOCRETRIEVAL

Uvádí, zda klient automaticky zobrazuje dokument, když kritériím dotazu odpovídá pouze jeden dokument. Tato funkce znamená, že pro dotazy, jejímž kritériím bude vyhovovat pouze jeden dokument, můžete nastavit systém tak, aby vynechal webovou stránku seznamu

dokumentů a dokument zobrazil, aniž by uživatel musel provést nějakou akci. Předvolená hodnota je 0 (nula) a znamená, že aplikace ODWEK zobrazí webovou stránku seznamu dokumentů i tehdy, pokud kritériím dotazu vyhovuje pouze jeden dokument.

Důležité: Nepovolujte automatické načítání dokumentů, jestliže plánujete používat prohlížeč Microsoft Internet Explorer. IBM navrhuje, abyste uvedli AUTODOCRETRIEVAL=0 v každé sekci prohlížeče, kterou definujete pro Internet Explorer.

Tento parametr má globální rozsah a můžete jej v sekci DEFAULT BROWSER zadat pouze jednou.

Tento parametr je volitelný.

Příklad:

```
[DEFAULT BROWSER]
AUTODOCRETRIEVAL=1
```

EMAILVIEWING

Když uživatel načítá dokument EMAIL ze serveru OnDemand, hodnota tohoto parametru určuje, jakou akci - pokud vůbec nějakou - aplikace ODWEK provede, než dokument odešle klientovi.

Tento parametr můžete nastavit na jednu z následujících hodnot:

NATIVE Aplikace ODWEK rozbaluje a dekomprimuje dokumenty EMAIL ze serveru OnDemand.

Poznámka: Jestliže uvádíte EMAIL=NATIVE, ověřte, zda typ obsahu MIME uvádí prohlížeč, který chcete používat. Podrobnosti viz téma “[MIMETYPES]” na stránce 22.

HTML Aplikace ODWEK konvertuje dokumenty EMAIL na dokumenty HTML. Je to předvolená hodnota.

Tento parametr má globální rozsah a můžete jej v sekci DEFAULT BROWSER zadat pouze jednou. Když používáte funkci Retrieve Document, můžete zadanou akci potlačit parametrem _email.

Tento parametr je volitelný.

Příklad:

```
[DEFAULT BROWSER]
EMAILVIEWING=HTML
```

ENCRYPTCOOKIES

Určuje, zda aplikace ODWEK šifruje objekty cookie, které jsou zasílány do webového prohlížeče. Předvolená hodnota je 0 (nula) a znamená, že objekty cookie se nešifrují. Chcete-li šifrovat všechny objekty cookie, které jsou odesílány do prohlížeče, zadejte 1 (jedna).

Tento parametr má globální rozsah a můžete jej v sekci DEFAULT BROWSER zadat pouze jednou.

Tento parametr je volitelný.

Příklad:

```
[DEFAULT_BROWSER]
ENCRYPTCOOKIES=1
```

ENCRYPTURL

Určuje, zda aplikace ODWEK šifruje hodnoty `server`, `userid`, `password` a `docid`, které jsou obsaženy v adrese URL, jež se odesílá do prohlížeče. Předvolená hodnota je 0 (nula) a znamená, že tyto hodnoty se nešifrují. Chcete-li tyto hodnoty šifrovat, zadejte 1 (jedna).

Tento parametr má globální rozsah a můžete jej v sekci DEFAULT BROWSER zadat pouze jednou.

Tento parametr je volitelný. Jestliže však musíte používat metodu GET pro přenos formulářových parametrů a hodnot mezi prohlížečem a HTTP serverem, pak můžete tyto hodnoty šifrovat zadáním ENCRYPTURL=1. Další informace o atributu metody příznaku formuláře najdete v tématu “Zabezpečení serveru a dat” na stránce 6.

Příklad:

```
[DEFAULT_BROWSER]
ENCRYPTURL=1
```

FOLDERDESC

Uvádí, zda se popis složek zobrazuje na pravé straně jména složky na stránce pro výběr složky. Předvolená hodnota je 0 (nula) a znamená, že popis složky se nebude zobrazovat. Chcete-li zobrazovat popis složky, zadejte 1 (jedna). Pokud tento parametr není uveden nebo pokud hodnota není definována nebo rozpoznána, popis složky se nebude zobrazovat.

Tento parametr má globální rozsah, pokud není potlačen v sekci prohlížeče (viz téma “[browser]” na stránce 35). Tento parametr můžete zadat pouze jednou v sekci DEFAULT BROWSER.

Tento parametr je volitelný.

Příklad:

```
[DEFAULT_BROWSER]
FOLDERDESC=1
```

LINEVIEWING

Když uživatel načítá dokument Line Data ze serveru OnDemand, hodnota tohoto parametru určuje, jakou akci - pokud vůbec nějakou - aplikace ODWEK provede, než dokument odešle klientovi.

Tento parametr můžete nastavit na jednu z následujících hodnot:

- | | |
|---------------|---|
| APPLET | Aplikace ODWEK konvertuje dokumenty Line Data pro zobrazení pomocí appletu Line Data (předvolba). |
| ASCII | Aplikace ODWEK konvertuje dokumenty Line Data na ASCII text. |
| NATIVE | Aplikace ODWEK rozbaluje a dekomprimuje dokumenty Line Data ze serveru OnDemand. |

Poznámka: Jestliže uvádíte LINEVIEWING=NATIVE, ověřte, zda typ obsahu MIME uvádí prohlížeč, který chcete používat. Podrobnosti viz téma “[MIMETYPES]” na stránce 22.

Tento parametr má globální rozsah a můžete jej v sekci DEFAULT BROWSER zadat pouze jednou. Když používáte funkci Retrieve Document, můžete zadanou akci potlačit parametrem `_line`.

Tento parametr je volitelný.

Příklad:

```
[DEFAULT BROWSER]
LINEVIEWING=APPLET
```

MAXHITS

Maximální počet položek vrácených do seznamu dokumentů, neohledně na počet položek, které odpovídají dotazu.

Poznámky:

1. Seznam dokumentů obsahuje položky, které odpovídají dotazu v pořadí, v jakém byly položky zavedeny do databáze.
2. Aplikace ODWEK používá první zadanou hodnotu ke zjištění počtu položek, který se vrací do seznamu dokumentů:
 - a. Pro funkci Document Hit List je to hodnota v poli Maximum Hits (uvedená na straně Permissions příslušné složky). Tato hodnota potlačí všechny jiné hodnoty.
 - b. Pro funkce Document Hit List a Print Document je to hodnota parametru `_max_hits`, pokud je zadána pro nějakou funkci. Hodnota parametru `_max_hits` potlačí parametr MAXHITS.
 - c. Hodnota parametru MAXHITS, pokud je zadána.
 - d. Jestliže není zadána žádná z výše uvedených hodnot, aplikace ODWEK vrátí do seznamu dokumentů maximálně 200 položek.

Tento parametr má globální rozsah a můžete jej v sekci DEFAULT BROWSER zadat pouze jednou.

Tento parametr je volitelný.

Příklad:

```
[DEFAULT BROWSER]
MAXHITS=200
```

NOLINKS

Určuje, zda seznam dokumentů obsahuje ovládací prvky pro prohlížení dokumentů. Jestliže je to povoleno, aplikace ODWEK přidá ovládací prvek ke každému dokumentu. Pokud si uživatel chce prohlédnout dokument, musí použít ovládací prvek. Předvolená hodnota je 0 (nula) a znamená, že uživatel musí použít textový odkaz, pokud chce zobrazit dokument.

Důležité: Jestliže používáte prohlížeč Microsoft Internet Explorer musíte nastavit NOLINKS=0. IBM navrhuje, abyste uvedli NOLINKS=0 v každé sekci prohlížeče, kterou definujete pro Internet Explorer.

Tento parametr má globální rozsah a můžete jej v sekci DEFAULT BROWSER zadat pouze jednou.

Tento parametr je volitelný.

Příklad:

```
[DEFAULT BROWSER]
NOLINKS=1
```

ODApplet.jre.path.IE

Viz téma “Java prohlížeč Line Data Viewer” na stránce 48.

ODApplet.jre.path.NN

Viz téma “Java prohlížeč Line Data Viewer” na stránce 48.

ODApplet.jre.version

Viz téma “Java prohlížeč Line Data Viewer” na stránce 48.

ODApplet.version

Viz téma “Java prohlížeč Line Data Viewer” na stránce 48.

SERVERPRINT

Určuje, zda seznam dokumentů obsahuje ovládací prvky pro zaslání dokumentů do serverové tiskárny. Jestliže je to povoleno, aplikace ODWEK přidá ovládací prvek ke každému dokumentu. Předvolená hodnota je 0 (nula) a znamená, že uživatelé musí nejprve otevřít dokument, a teprve pak jej mohou odeslat do serverové tiskárny.

Poznámky:

1. Chcete-li použít tisk na serveru, musí být na serveru OnDemand definována alespoň jedna serverová tiskárna.
2. Uživatelé mohou nebo nemohou tisknout dokumenty, což závisí na oprávněních Print ve skupině aplikací OnDemand.

Tento parametr má globální rozsah a můžete jej v sekci DEFAULT BROWSER zadat pouze jednou.

Tento parametr je volitelný.

Příklad:

```
[DEFAULT BROWSER]
SERVERPRINT=1
```

SERVERPRINTERS

Používá se k zadávání typů serverových tiskáren, které si uživatel může vybírat. Existují tři typy serverových tiskáren:

- P** Serverová tiskárna
- I** Serverová tiskárna s informacemi
- F** Serverový fax

Můžete zadat typy od nuly do tří, v seznamu oddělovaném čárkami.

Následující příklad uvádí, že uživatel může vybírat serverové tiskárny a serverové faxy:

```
[DEFAULT BROWSER]
SERVERPRINTERS=P,F
```

SHOWDOCLOCATION

Když generujete oddělovaný ASCII výstup namísto HTML (viz téma Dodatek H, “Žádný HTML výstup”, na stránce 131), určuje, zda se úložiště dokumentu bude zobrazovat ve výstupu. Podrobnosti viz téma “Seznam nalezených dokumentů” na stránce 133. Předvolená hodnota je 0 (nula) a znamená, že úložiště se nebude zobrazovat ve výstupu.

Poznámka: Chcete-li zobrazit úložiště, musíte také nastavit vlastnost Display Document Location ve složce OnDemand.

Tento parametr má globální rozsah a můžete jej v sekci DEFAULT BROWSER zadat pouze jednou.

Tento parametr je volitelný.

Příklad:

```
[DEFAULT BROWSER]
SHOWDOCLOCATION=1
```

VIEWNOTES

Určuje, zda lze zobrazit anotace k dokumentům. Pokud je to povoleno, aplikace ODWEK umístí ovládací prvek pro zobrazení anotací vedle každého dokumentu v seznamu dokumentů. Předvolená hodnota je 0 (nula) a znamená, že anotace nelze prohlížet.

Poznámka: Uživatelé mohou nebo nemohou prohlížet anotace do dokumentů, což závisí na oprávněních Annotation ve skupině aplikací OnDemand.

Tento parametr má globální rozsah a můžete jej v sekci DEFAULT BROWSER zadat pouze jednou.

Tento parametr je volitelný.

Příklad:

```
[DEFAULT BROWSER]
VIEWNOTES=1
```

[browser]

Můžete zadat volby pro specifické prohlížeče, které se používají ve vaší organizaci. Parametry, které zadáte do sekce browser, potlačí parametry ze sekce DEFAULT BROWSER v souboru ARSWWW.INI. (Použijí se parametry, které zadáte do sekce DEFAULT BROWSER, pokud je nezadáte do sekce browser.)

Poznámky:

1. Hlavička sekce musí obsahovat řetězec, který označuje prohlížeč, pro který chcete zadat volby. Aplikace ODWEK zjistí hodnotu proměnné prostředí HTTP_USER_AGENT, aby zjistila, který prohlížeč se používá. Aplikace ODWEK pak prohledá soubor ARSWWW.INI, zda v něm není sekce browser, která odpovídá hodnotě. Pokud nebude nalezena žádná sekce browser, aplikace ODWEK pak prohledá soubor ARSWWW.INI, zda neobsahuje jednu z následujících sekcí:

```
[browser version(major.minor)/platform]
```

```
[browser version(major.minor)]
```

```
[browser version(major)]
```

```
[browser]
```

```
[DEFAULT BROWSER]
```

Aplikace ODWEK používá volby z první sekce, která odpovídá hodnotě.

2. Jako hodnotu browser můžete zadat IE nebo Netscape.
3. Jako hodnotu platform můžete uvést WinNT nebo Unix.

Sekce browser má pro zadaný prohlížeč globální rozsah. Pro každý prohlížeč zadejte pouze jednu sekci browser. Měli byste uvádět pouze ty parametry, které potřebujete potlačit oproti sekci DEFAULT BROWSER.

Tato sekce je volitelná.

Tato sekce může obsahovat stejné parametry, které jsou definovány pro předvolený prohlížeč. Viz téma “[DEFAULT BROWSER]” na stránce 28.

Příklady:

```
[IE 5]
AUTODOCRETRIEVAL=0
NOLINKS=0
```

```
[Netscape 4.7]
AUTODOCRETRIEVAL=1
NOLINKS=1
```

[DEBUG]

Sekce DEBUG obsahuje volby, které můžete použít pro řešení problémů, jež máte vy a jiné osoby ve vaší organizaci s používáním aplikace ODWEK. Sekce DEBUG musí být první spustitelný příkaz v souboru arswww.ini.

Sekce DEBUG má globální rozsah a můžete ji zadat do souboru ARSWWW.INI pouze jednou.

Tato sekce je volitelná.

Tato sekce může obsahovat následující parametry:

LOG

Umožňuje aplikaci ODWEK zapisovat zprávy a jiné informace o programech do souboru protokolu. (Soubor protokolu má jméno ARSWWW.LOG.)

Tento parametr má globální rozsah a můžete jej v sekci DEBUG zadat pouze jednou.

Tento parametr je volitelný. Předvoleno je, že aplikace ODWEK nezapisuje zprávy do souboru protokolu. Chcete-li zprávy protokolovat, zadejte hodnotu 1 (jedna).

LOGDIR

Určuje adresář, do kterého aplikace ODWEK zapisuje soubor ARSWWW.LOG, jestliže je protokolování povoleno parametrem LOG.

Tento parametr má globální rozsah a můžete jej v sekci DEBUG zadat pouze jednou.

Tento parametr je volitelný. Je předvoleno, že pokud je povoleno protokolování, aplikace ODWEK zapisuje soubor protokolu do adresáře /QIBM/UserData/OnDemand/WWW/LOG.

Příklad:

```
[DEBUG]
LOGDIR=/QIBM/UserData/OnDemand/WWW/LOG
LOG=1
```

Příklad souboru ARSWWW.INI

S produktem se dodává vzorový konfigurační soubor ARSWWW.INI. Vzorový konfigurační soubor nastavuje nejčastěji používané předvolené hodnoty pro servery, webové prohlížeče nebo jiné prohlížeče.

```
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;; Server Configuration   ;;;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
```

```

;[DEBUG]
;log=1
;logdir=/QIBM/UserData/OnDemand/www/logs

[@SRV@_<host alias>];
HOST=<host name>;
PORT= ;
PROTOCOL= ;

[@SRV@_myiSeries.mycompany.com]
HOST=myiSeries.mycompany.com
PORT=1450
PROTOCOL=0

[CONFIGURATION]
CodePage=37
Language=ENU
TemplateDir=/QIBM/UserData/OnDemand/www/SAMPLES
ImageDir=/IMAGES/
AppletDir=/applets/
CacheDir=/QIBM/UserData/OnDemand/www/
CacheSize=0
CacheMinThreshold=0
CacheMaxThreshold=0
CacheDocs=0
CacheUserIDs=web,demo,mstephens

[SECURITY]
SERVERACCESS=

[AFP2HTML]
InstallDir=/QIBM/UserData/OnDemand/www/bin
ConfigFile=/QIBM/UserData/OnDemand/www/bin/afp2html.ini
UseExecutable=1

[AFP2PDF]
InstallDir=/QIBM/UserData/OnDemand/www/bin
ConfigFile=/QIBM/UserData/OnDemand/www/bin/afp2pdf.ini
UseExecutable=1

[MIMETYPES]
BMP=image/IBM_OnDemand
GIF=image/IBM_OnDemand
JFIF=image/IBM_OnDemand
PCX=image/IBM_OnDemand
TIFF=image/IBM_OnDemand
PNG=image/IBM_OnDemand
PDF=application/pdf
AFP=application/afp
LINE=application/line
EMAIL=text/html
META=application/unknown

[ATTACHMENT IMAGES]
TXT=userText.gif
BMP=userBitMap.gif
GIF=userGIF.gif

[NO HTML]
BEGIN=[
END=]
SEPARATOR=^

;;;;;;;;;;;;;;;;;;;;;;;;;
;;; Default Browser   ;;;
;;;;;;;;;;;;;;;;;;;;;;;;;

```

```
[DEFAULT_BROWSER]
```

```
;AfpViewing=[ascii,html,native,pdf,plugin,xenos]  
AfpViewing=plugin  
;LineViewing=[ascii,applet,native]  
LineViewing=applet  
;EmailViewing=[html,native]  
EmailViewing=html  
;MetaViewing=[xenos,native]  
MetaViewing=native  
NoLinks=1  
ViewNotes=1  
AddNotes=1  
ServerPrint=1  
ServerPrinters=P  
AutoDocRetrieval=1  
MaxHits=200  
ShowDocLocation=1
```

```
[IE]
```

```
NoLinks=0  
AddExtension=1  
AddFieldsToDocid=0
```

Váš další krok

Jakmile nainstalujete software ODWEK a nakonfigurujete soubor ARSWWW.INI, měli byste nakonfigurovat vzorové aplikace. Viz téma Kapitola 3, “Konfigurace vzorových aplikací”, na stránce 39.

Kapitola 3. Konfigurace vzorových aplikací

Tato kapitola vysvětluje, jak lze přizpůsobit vzorové aplikace, které se dodávají s produktem ODWEK:

- LOGON.HTM. Tato aplikace podporuje uživatele, kteří mají oprávnění přistupovat k několika složkám. Každý uživatel je definován na serveru OnDemand. Po přihlášení k serveru aplikace ODWEK ukáže uživateli seznam složek, které uživatel může otevřít.
- CREDIT.HTM. Tato aplikace podporuje náhodné použití serveru OnDemand. Uživatel zadá vyhledávací kritéria pro určitou složku. Jméno serveru OnDemand, ID uživatele a heslo, jméno složky a pole složky jsou zakódovány do aplikace. Téma "CREDIT.HTM" obsahuje pokyny pro přizpůsobení této aplikace.

Poté, co upravíte vzorové aplikace, publikujte URL všech souborů, aby se k nim uživatelé mohli připojit a přistupovat k serveru OnDemand. Každý příklad vyžaduje odlišnou úroveň přizpůsobení. K přizpůsobení jedné ze vzorových aplikací se dodává kompletní návod. Návod použijte jako vodítko pro případné potřebné přizpůsobení jiných aplikací.

Poznámka: Kromě přizpůsobení vzorových aplikací IBM doporučuje, abyste přizpůsobili soubor TEMPLATE.HTM pro svou organizaci. Soubor TEMPLATE.HTM obsahuje uživatelsky definovaný obsah, který aplikace ODWEK používá pro zobrazení webových stránek. Důležité informace o přizpůsobení tohoto souboru najdete v tématu "TEMPLATE.HTM" na stránce 40.

LOGON.HTM

1. Zkopírujte soubor logon.htm z instalačního adresáře /QIBM/ProdData/OnDemand/www.samples) do kořenového adresáře dokumentů na HTTP serveru.
2. Pro program CGI ověřte, zda soubor logon.htm obsahuje následující řádky:

```
<h4>Zadejte prosím přihlašovací informace:</h4>
<FORM METHOD=POST ACTION="/arswww.cgi">
```
3. Pro servlet ověřte, zda soubor logon.htm obsahuje následující řádek:

```
<FORM METHOD=POST ACTION="/ArsWWWServlet">
```

CREDIT.HTM

Přizpůsobte vzorovou aplikaci CREDIT.HTM tak, že vytvoříte kopii souboru pro každou složku, k nimž mají uživatelé přistupovat. Jméno souboru by mělo být stejné jako jméno složky.

1. Upravte soubor CREDIT.HTM. (V rámci předvolby je tento soubor umístěn v adresáři /QIBM/UserData/OnDemand/WWW/SAMPLES.)
2. Změňte nebo vymažte obrázek pozadí zadaný v příkazu <body> (řádek 11).
3. Pokud chcete, můžete změnit barvu pozadí zadanou v příkazu <body> (řádek 11).
4. Změňte nebo vymažte obrázek produktu zadaný v příkazu (řádek 12).
5. Nahaďte jméno složky uvedené v příkazu <h1> (řádek 15).
6. Nahaďte text zadaný v příkazech <p> (řádky 17 až 25). Zadejte obecné pokyny pro uživatele.
7. Nahaďte jméno adresáře CGI-BIN uvedené v příkazu <FORM> (řádek 29). Zadejte jméno adresáře CGI-BIN, které obsahuje programy a soubory ODWEK na HTTP serveru.

8. Nahraďte hodnotu zadanou v příkazu `<input>` (řádek 30). Toto je řetězec dělený čárkami a obsahující jména zobrazovaných polí složky.
9. Nahraďte hodnotu zadanou v příkazu `<input>` (řádek 31). Toto je jméno složky.
10. Nahraďte hodnotu zadanou v příkazu `<input>` (řádek 33). Toto je maximální počet položek zobrazovaných v seznamu dokumentů, nehledě na počet položek, které odpovídají dotazu.
11. Nahrazuje jméno serveru zadané v příkazu `<input>` (řádek 35). Toto je jméno serveru OnDemand, s nímž má aplikace ODWEK komunikovat. Zadané jméno serveru je `gunnar`.
12. Jestliže chcete třídit položky v seznamu dokumentů, ověřte hodnotu zadanou v příkazu `<input>` (řádek 36). V opačném případě vymažte řádek 36.
13. Jestliže chcete třídit položky v seznamu dokumentů, ověřte hodnotu zadanou v příkazu `<input>` (řádek 37). V opačném případě vymažte řádek 37.
14. Nahraďte hodnotu zadanou v příkazu `<input>` (řádek 38). Toto je ID uživatele na serveru OnDemand. ID uživatele, které zadáte, musí mít oprávnění pro otevření složky a pro přístup k datům skupiny aplikací.
15. Chcete-li, můžete také změnit jméno souboru šablony uvedené v příkazu `<input>` (řádek 39). Server OnDemand používá soubor šablony ke generování dalších webových stránek. Zadané jméno souboru šablony je `template.htm`.
16. Upravte řádky 40 až 43 pro první pole pro vyhledávání složek.
 - a. Napište jméno pole složky do příkazu ``.
 - b. Hodnotu zadanou v poli jména v příkazu `<input>` nahraďte jménem pole skutečné složky.
 - c. Hodnotu zadanou v poli hodnoty v příkazu `<input>` nahraďte předvolenou vyhledávací hodnotou.
17. Okopírujte řádky 40 až 43 a zopakujte krok 16 pro každé další vyhledávací pole složky.
18. Uložte změny a zavřete textový editor.

TEMPLATE.HTM

Soubor TEMPLATE.HTM je předvolený soubor šablony používaný aplikací ODWEK pro generování webových stránek podle různých funkcí produktu (například přihlášení). Tento soubor byste měli nahradit souborem, který obsahuje uživatelsky definovaný obsah. Avšak soubor šablony musí obsahovat následující řádku komentáře HTML:

```
<!-- - - AOI# Marker - - -->
```

Umístění řádku komentáře určuje, kam aplikace ODWEK umísťuje svůj výstup. Všechny řádky uvedené nad řádkem komentáře se zapíší před výstup generovaný aplikací ODWEK. Všechny řádky uvedené pod řádkem komentáře se zapíší za výstup generovaný aplikací ODWEK.

Je předvoleno, že soubor šablony je umístěn v adresáři nazvaném parametrem `TEMPLATEDIR` v souboru `ARSWWW.INI`. Podrobnosti viz téma “`TEMPLATEDIR`” na stránce 19.

Váš další krok

Jakmile nainstalujete software ODWEK, konfigurovaný soubor `ARSWWW.INI` a konfigurované vzorové aplikace, měli byste nainstalovat webové prohlížeče na uživatelské pracovní stanice. Viz téma Kapitola 4, “Instalace webových prohlížečů”, na stránce 41.

Kapitola 4. Instalace webových prohlížečů

Přehled

IBM poskytuje prohlížeče pro standardní typy dokumentů, které lze načítat ze serveru OnDemand. Požadavky na instalaci se liší podle prohlížečů, které pracovníci vaší organizace potřebují používat.

- Pro zobrazení dokumentů Line Data IBM doporučuje, abyste používali applet Line Data. Applet Line Data je uložen na HTTP serveru. Jakmile povolíte použití appletu Line Data, applet se automaticky zavede do paměti pracovní stanice, když si uživatel začne prohlížet dokument Line Data. Ověřte, aby parametr LINEVIEWING v souboru ARSWWW.INI uváděl prohlížeč, který vaši uživatelé budou používat.
- Pro zobrazení dokumentů AFP můžete použít webový prohlížeč IBM OnDemand AFP, applet AFP2HTML nebo prohlížeč Adobe Acrobat.
 - Aby bylo možné zobrazovat dokumenty AFP webových prohlížečem IBM OnDemand AFP, uživatelé jej musí mít instalovaný na svých pracovních stanicích.
 - Aby bylo možné zobrazit dokumenty AFP appletem AFP2HTML, administrátor musí instalovat a konfigurovat službu AFP2WEB Transform na HTTP serveru a konfigurovat soubor ARSWWW.INI. Applet AFP2HTML se ukládá na HTTP server. Jakmile administrátor povolí použití appletu AFP2HTML, applet se automaticky zavede do paměti na pracovní stanici, když si uživatel zvolí zobrazení dokumentu AFP.
 - Aby bylo možné zobrazit dokumenty AFP prohlížečem Adobe Acrobat, administrátor musí instalovat a konfigurovat službu AFP2PDF Transform na HTTP serveru a konfigurovat soubor ARSWWW.INI. Poté, co administrátor povolí použití převodu, je předvoleno, že prohlížeč se bude pokoušet spustit program Adobe Acrobat Viewer, když si uživatel zvolí zobrazení dokumentu AFP. Uživatel musí získat a nainstalovat Adobe Acrobat na pracovní stanici.

Ověřte, aby parametr AFPVIEWING v souboru ARSWWW.INI uváděl prohlížeč, který vaši uživatelé budou používat.

- Chcete-li prohlížet soubory ve formátu BMP, GIF, JPEG, PCX a TIFF, IBM doporučuje, aby vaši uživatelé instalovali Image Web Viewer IBM OnDemand na své pracovní stanice. Pokud to neudělají, měli by použít nějaký jiný prohlížeč, který pracuje s těmito typy dokumentů. (Většina webových prohlížečů má například integrované prohlížeče, které dokáží zobrazit soubory GIF a JPEG.) Jestliže se vaši uživatelé rozhodnou používat nějaký jiný druh prohlížeče, zajistěte, aby administrátor změnil pro tyto soubory předvolený typ obsahu MIME. Ověřte, že parametry v sekci MIMETYPES v souboru ARSWWW.INI uvádějí prohlížeče, které budou vaši uživatelé používat.

Poznámky:

1. Chcete-li zobrazovat jiné typy dat, možná budete muset instalovat jiné prohlížeče. Například chcete-li zobrazit dokumenty PDF, které se načítají ze serveru OnDemand, IBM doporučuje, abyste získali a nainstalovali prohlížeč Adobe Acrobat pro webové prohlížeče, které se používají ve vaší organizaci.
2. Pro prohlížení dokumentů PDF je zapotřebí soubor nppdf32.dll umístěný v adresáři programů typu plugin webového prohlížeče. Pro Internet Explorer by měl být v adresáři \Program Files\Internet Explorer\PLUGINS. Pro Netscape by měl být v adresáři \Program Files\Netscape\Communicator\Program\Plugins. Jestliže soubor není v adresáři prohlížeče, budete muset znovu nainstalovat software Adobe.

Požadavky

Prohlížeče, které dodává IBM, vyžadují Netscape Navigator 4.7 či novější nebo Microsoft Internet Explorer 5.5 či novější.

Aplikace ODWEK vyžaduje schopnost pro zápis dat objektů cookie na PC. Zajistěte, aby vaši uživatelé konfigurovali své prohlížeče pro příjem objektů cookie.

IBM poskytuje dvě verze dálkového datového prohlížeče Java v adresáři appletů:

ODLineDataViewer.jar je starý prohlížeč Line Data Java, který vyžaduje podporu Java ve webovém prohlížeči. Podpora Java je nejpravděpodobněji poskytována na Java Virtual Machine (JVM). **Poznámka:** Pro Microsoft Internet Explorer bude uživatel možná muset nainstalovat JVM tím, že ve webovém prohlížeči zvolí vlastní (Custom) instalaci.

ODLineDataViewer2.jar je nový dálkový datový Java prohlížeč, který vyžaduje verzi 1.4.1 nebo vyšší verzi Java pluginu. Nový Java prohlížeč Line Data Viewer nepoužívá podporu Javy ve webovém prohlížeči. Uživatel musí instalovat Java plugin na PC, aby bylo možné používat nový Java prohlížeč Line Data Viewer. Důležité informace o konfiguraci najdete v tématu "Java prohlížeč Line Data Viewer" na stránce 48.

Předvolená instalace produktu bude používat starý Java prohlížeč Line Data Viewer.

Prohlížeč se musí spouštět v systému Windows 2000, Windows XP nebo Windows Server 2003 a vyžaduje následující hardware a software:

- Fyzické připojení k síti, například síťový adaptér Token Ring nebo Ethernet.
- Podpora Windows TCP/IP.
- Minimálně 256 MB RAM.
- Procesor Pentium nebo kompatibilní s Pentiem s kmitočtem 800 MHz nebo vyšším.
- Obrazovka SVGA a adaptér s minimálním rozlišením 800 x 600.
- Minimálně 20 MB volného diskového prostoru pro prohlížení dokumentů.
- Přibližně 3 MB na každé pracovní stanici, která potřebuje IBM OnDemand AFP Web Viewer, a 2 MB na každé pracovní stanici, která potřebuje IBM OnDemand Image Web Viewer.

Instalace

Poznámka: Jestliže plánujete, že pomocí AFP Web Viewer budete distribuovat uživatelem definované soubory, měli byste nakonfigurovat instalační soubor AFP Web Viewer tak, aby zahrnoval uživatelsky definované soubory, a to před tím, než uživatelé začnou instalovat AFP Web Viewer. Další informace najdete v tématu "Distribuce uživatelsky definovaných souborů" na stránce 43.

Prohlížeče dodané od IBM se instalují pomocí samorozbalovacích souborů. Tyto soubory si uživatel musí stáhnout do systému Windows a spustit je, čímž nainstaluje příslušný prohlížeč. Jestliže uživatel používá webový prohlížeč, když probíhá instalace, pak uživatel musí webový prohlížeč zavřít a znovu spustit - teprve pak bude možné použít prohlížeč. Následující soubory prohlížeče lze nalézt v adresáři /QIBM/ProdData/OnDemand/www/plugins:

- afpplgus.exe - IBM OnDemand AFP Web Viewer - pouze anglicky
- afpplgin.exe - IBM OnDemand AFP Web Viewer - všechny jazyky včetně podpory DBCS
- afpplgin.zip - IBM OnDemand AFP Web Viewer - formát Zip pro všechny jazyky zahrnuje podporu DBCS
- imgplgin.exe - IBM OnDemand Image Web Viewer - všechny jazyky

Během instalačního procesu se zkopíruje prohlížeč a přidružené soubory do adresářů vybraných uživatelem. Prohlížeč AFP Web Viewer vyžaduje přibližně 3 MB prostoru na pracovní stanici. Prohlížeč Image Web Viewer vyžaduje přibližně 2 MB prostoru na pracovní stanici. Připomeňte svým uživatelům, že musí restartovat své webové prohlížeče, pokud je měli při instalaci spuštěné.

Poznámka: Instalační program nainstaluje prohlížeče buď jako programy typu plugin, nebo jako ovládací prvky ActiveX. Pokud je na pracovní stanici instalován Internet Explorer, pak instalační program nainstaluje ovládací prvky ActiveX; pokud je na pracovní stanici nainstalován Netscape, pak instalační program nainstaluje programy typu plugin. Pokud na pracovní stanici máte instalovaný Internet Explorer a Netscape, instalační program nainstaluje ovládací prvky ActiveX pro Internet Explorer a programy typu plugin pro Netscape.

Distribuce uživatelsky definovaných souborů

Uživatelsky definované soubory můžete distribuovat se softwarem IBM OnDemand AFP Web Viewer, který dodává IBM. Například předpokládejme, že někdo ve vaší organizaci vytvoří soubory fontů AFP pro dokumenty, které jsou uloženy na serveru OnDemand. Takové soubory fontů můžete distribuovat se softwarem AFP Web Viewer. Tímto způsobem, když si uživatel bude prohlížet dokument AFP, dokument se zobrazí s příslušnými fonty.

Chcete-li se softwarem AFP Web Viewer distribuovat uživatelsky definované soubory, musíte soubory zabalit do instalačního souboru a umístit instalační soubor do sdíleného umístění. Když uživatel spustí instalační soubor, program Setup automaticky nainstaluje software AFP Web Viewer a uživatelsky definované soubory na pracovní stanici uživatele.

Se softwarem AFP Web Viewer můžete distribuovat následující typy uživatelem definovaných souborů:

- Soubory fontů AFP. Tyto soubory se zkopírují do podadresáře FONT v cílovém adresáři softwaru AFP Web Viewer na pracovní stanici.
- Soubory fontů Adobe Type 1. Tyto soubory se zkopírují do adresáře zadaného uživatelem a instalovaného v ATM programem Setup.
- Soubory fontů TrueType. Tyto soubory se kopírují do adresáře Windows FONTS a instalují do systému Windows programem Setup.
- Různé uživatelsky definované soubory. Tyto soubory se kopírují do cílového adresáře softwaru AFP Web Viewer na pracovní stanici uživatele.

Poznámka: Program Setup kopíruje uživatelsky definované soubory do pracovní stanice po souborech AFP Web Viewer, které dodá IBM. Jestliže uživatelsky definovaný soubor pojmenujete stejně jako soubor dodaný od IBM, uživatelsky definovaný soubor nahradí soubor od IBM. Této skutečnosti můžete využít například pro distribuci aktualizovaného souboru FLDPORT2.INI nebo pro distribuci souborů fontů IBM AFP, které byly upraveny ve vaší organizaci.

Další informace o konfiguraci a distribuci softwaru AFP Web Viewer jsou uvedeny v následujících tématech:

- Instalace souborů AFP Web Viewer dodaných od IBM.
- Přidání podadresářů pro uložení uživatelsky definovaných souborů.
- Uložení uživatelsky definovaných souborů do adresářů.
- Konfigurace souborů fontů.
- Vytvoření instalačního souboru AFP Web Viewer.
- Instalace softwaru AFP Web Viewer na pracovní stanici uživatele.

Instalace souborů AFP Web Viewer

Většina zákazníků používá jeden ze dvou způsobů distribuce souborů prohlížeče ze serveru, a to podle toho, zda plánují distribuovat uživatelsky definované soubory se softwarem AFP Web Viewer:

- Standardní instalace. Používá se pro distribuci souborů AFP Web Viewer dodaných od IBM a pro přípravu na distribuci uživatelsky definovaných souborů s webovým prohlížečem AFP. Když administrátor nainstaluje software ODWEK na HTTP server, instalační soubory pro prohlížeče se uloží do adresáře na serveru. Pro každý prohlížeč a soubor archivu ZIP pro AFP Web Viewer by měl existovat jeden instalační soubor (EXE). Administrátor obvykle přesune instalační soubory do veřejného adresáře na serveru a vytvoří webovou stránku s odkazy na soubory. Uživatel si prohlížeč nainstaluje stažením webové stránky do svého webového prohlížeče a klepnutím na odkaz na příslušný instalační soubor.
- Přizpůsobená instalace AFP Web Viewer. Používá se pro distribuci uživatelsky definovaných souborů v softwaru AFP Web Viewer.
 1. Nastavení serveru pro standardní instalaci.
 2. Než někteří uživatelé skutečně nainstalují prohlížeč, nejprve získejte kopii archivního souboru ZIP AFP Web Viewer.
 3. Soubory z archivního souboru ZIP rozbalte do prázdného adresáře.
 4. Do pracovního adresáře přidejte podadresáře a do nich uložte uživatelsky definované soubory. Podrobnosti najdete v tématu “Přidávání podadresářů” a v tématu “Ukládání uživatelsky definovaných souborů” na stránce 45.
 5. Jestliže distribuujete uživatelsky definované soubory fontů Adobe Type 1, pak vytvořte konfigurační soubor fontů. Podrobnosti viz téma “Konfigurace souborů fontů” na stránce 45.
 6. Poté, co nakonfigurujete všechny adresáře a soubory, vytvořte samorozbalovací soubor EXE pro distribuci. Podrobnosti viz téma “Vytvoření instalačního souboru AFP Web Viewer” na stránce 46.
 7. Soubor EXE dodaný od IBM pro standardní instalaci nahraďte samorozbalovacím souborem EXE, který jste vytvořili.
 8. Poté, co administrátor provede kroky 1 až 7, uživatelé mohou instalovat software AFP Web Viewer a uživatelsky definované soubory tak, že si stáhnou webovou stránku do webového prohlížeče a kliknou na odkaz aktualizovaného instalačního souboru.

Přidávání podadresářů

Uživatelsky definované soubory, které plánujete distribuovat, musí být uloženy ve stromu podadresářů CUSTOM v hlavním instalačním adresáři klienta. Hlavní instalační adresář klienta můžete například pojmenovat \ONDEMAND\AFP32.

Konfigurace hlavního instalačního adresáře klienta pro uložení uživatelsky definovaných souborů:

1. Vytvořte adresář CUSTOM v hlavním instalačním adresáři klienta. Příklad:
`\ondemand\afp32\custom`

Poznámka: Adresář CUSTOM může obsahovat jiné¹ uživatelsky definované soubory, které chcete distribuovat svým uživatelům. Program Setup kopíruje soubory z tohoto adresáře do cílového adresáře AFP Web Viewer na pracovní stanici.

1. Jiné soubory fontů než AFP, soubory fontů Adobe Type 1 a soubory fontů Windows TrueType.

2. Do adresáře CUSTOM přidejte jeden nebo více z následujících podadresářů. To, jaké podadresáře byste měli přidat, závisí na typu uživatelsky definovaných souborů, které chcete distribuovat svým uživatelům.

- V adresáři CUSTOM vytvořte podadresář FONT, do kterého se budou ukládat soubory fontů AFP (typy souborů FNT a MAP). Příklad:

```
\ondemand\afp32\custom\font
```

Program Setup zkopíruje tyto soubory do adresáře AFP Web Viewer FONT na pracovní stanici.

- V adresáři CUSTOM vytvořte podadresář TYPEONE, do kterého se budou ukládat soubory fontů Adobe Type 1 (typy souborů PFB a PFM) a konfigurační soubor fontů. Příklad:

```
\ondemand\afp32\custom\typeone
```

Program Setup tyto soubory zkopíruje do adresáře zadaného uživatelem a nainstaluje fonty do ATM.

- V adresáři CUSTOM vytvořte podadresář TRUETYPE do kterého se budou ukládat soubory fontů Windows TrueType (typ souboru TTF). Příklad:

```
\ondemand\afp32\custom\truetype
```

Program Setup soubory zkopíruje z tohoto adresáře do adresáře Windows FONT a fonty nainstaluje do systému Windows.

Ukládání uživatelsky definovaných souborů

Poté, co rozbalíte instalační soubory od IBM do pracovního adresáře a vytvoříte adresáře CUSTOM, můžete uložit uživatelsky definované adresáře do jednotlivých podadresářů. Například soubory fontů Adobe Type 1 (typ souboru PFB a PFM), které chcete distribuovat svým uživatelům, zkopírujte do adresáře \ONDEMAND\AFP32\CUSTOM\TYPEONE.

Konfigurace souborů fontů

Jestliže plánujete distribuci uživatelsky definovaných souborů fontů Adobe Type 1 ke svým uživatelům, musíte provést následující kroky:

1. Uložte uživatelsky definované soubory fontů Type 1 (typy souborů PFB a PFM) do podadresáře TYPEONE v adresáři CUSTOM. Další informace najdete v tématu “Přidávání podadresářů” na stránce 44.
2. Vytvořte konfigurační soubor fontů Type 1. Níže je popsáno, jak lze vytvořit konfigurační soubor fontů Type 1.

Konfigurační soubor fontů Type 1 musí mít jméno ATM_INI.CFG a musí být uložen v podadresáři TYPEONE v adresáři CUSTOM. Další informace o distribučních adresářích najdete v tématu “Přidávání podadresářů” na stránce 44.

Každý záznam (řádek) v konfiguračním souboru fontů Type 1 označuje pouze jeden uživatelsky definovaný font Adobe Type 1, který chcete distribuovat svým uživatelům.

Formát záznamu je:

```
fontname=filename.PFM,filename.PFB
```

Parametr fontname je jméno fontu Type 1, jak se objevuje v seznamu fontů ovládacího panelu ATM, filename.PFM je jméno souboru PFM pro font a filename.PFB je jméno souboru PFB pro font. Následující příklad ukazuje konfigurační soubor fontů Type 1 se dvěma záznamy:

```
Courier,BOLD=coub.pfm,coub.pfb  
SonoranSansSerif_36,BOLDITALIC=c0a175z0.pfm,c0a175z0.pfb
```

První záznam v souboru označuje font Courier,BOLD a jeho PFM soubor fontu coub.pfm a PFB soubor fontu coub.pfb. Druhý záznam v souboru uvádí font SonoranSansSerif_36,BOLDITALIC a jeho PFM soubor fontu c0a175z0.pfm a PFB soubor fontu c0a175z0.pfb.

Když uživatel spustí instalační soubor AFP Web Viewer, který obsahuje uživatelsky definované soubory fontu Adobe Type 1, program Setup zpracuje soubory fontu následujícím způsobem:

1. Kopíruje všechny uživatelsky definované soubory fontů Adobe Type 1 (typy souboru PFB a PFM), které jsou v adresáři TYPEONE, do cílového adresáře. Cílový adresář zadává uživatel.
2. Ověřuje, zda pro každý font označený v konfiguračním souboru (ATM_INI.CFG) fontů Type 1 byly zkopírovány dva soubory. Jméno souborů zkopírovaných na pracovní stanici musí odpovídat jménům zadaným v konfiguračním souboru fontů.

Poznámka: Jestliže jména v souborech fontů zadaných v konfiguračním souboru fontů neodpovídají jménům souborů nakopírovaných do pracovní stanice, program Setup zobrazí varovnou zprávu a font nenainstaluje.

3. Přidává informace o cestě k souborům PFB a PFM pomocí cílového adresáře zadaného uživatelem.
4. Instaluje fonty do ATM.

Vytvoření instalačního souboru AFP Web Viewer

Poté, co dokončíte vytváření adresářů a ukládání souborů do adresářového stromu CUSTOM, musíte vytvořit instalační soubor, který obsahuje vaše uživatelsky definované soubory a soubory AFP Web Viewer od IBM. Instalační soubor se obvykle jmenuje Setup.exe.

Existují společnosti, které vytvářejí software pro balení souborů a aplikací do jednoho samorozbalovacího spustitelného souboru AFP Web Viewer pro distribuci. Například společnost InstallShield Software Corporation nabízí produkt nazvaný PackageForTheWeb.

Poznámka: Software dodávaný jinými společnostmi IBM nepodporuje.

Jakmile získáte balící software, spusťte jej a postupujte podle návodu k vytvoření instalačního souboru AFP Web Viewer, který obsahuje vaše uživatelsky definované soubory a soubory AFP Web Viewer od IBM.

Instalace softwaru AFP Web Viewer na pracovní stanici uživatele

Poté, co vytvoříte adresářový strom CUSTOM, vytvoříte instalační soubor AFP Web Viewer a nahradíte instalační soubor AFP Web Viewer na serveru, mohou uživatelé začít instalovat software AFP Web Viewer a uživatelsky definované soubory. Až příště nějaký uživatel klikne na odkaz na instalační soubor AFP Web Viewer ze serveru, program Setup nainstaluje software AFP Web Viewer na pracovní stanici uživatele a zkopíruje na pracovní stanici uživatele všechny uživatelsky definované soubory, které jste zabalili do instalačního souboru AFP Web Viewer.

Mapování fontů AFP

Fonty AFP, s nimiž byl dokument vytvořen, je nutné mapovat na fonty, které lze zobrazit pluginem AFP. ODWEK poskytuje soubory definic fontů, které mapují fonty IBM Core Interchange (pouze Latin) a kompatibilní na fonty TrueType. Soubory definic fontů a soubory mapování fontů jsou uloženy v podadresáři FONT, v němž je umístěn kód webového prohlížeče AFP.

Jestliže vaše dokumenty používají fonty, které nejsou definovány na webovém prohlížeči AFP, jestliže vy nebo jiní ve vaší organizaci upravili fonty IBM nebo jestliže vy nebo někdo jiný ve vaší organizaci vytvořili fonty AFP, pak musíte definovat fonty v souborech definic fontů, aby webový prohlížeč AFP mohl správně zobrazovat dokumenty. Podrobnosti o tom, jak mapovat fonty AFP, soubory definic fontů a jiné technické informace související s fonty AFP a TrueType najdete v publikaci *AFP Workbench Technical Reference*.

AFP Web Viewer

Následující nastavení mohou být použita z logických obrázků na serveru pro webový prohlížeč AFP.

- Barva pozadí. Jsou podporovány následující barvy. Nejsou podporovány žádné další barvy.
 - Zelený pruh (zobrazený s bílým pozadím)
 - Zelená
 - Červená
 - Žlutá
 - Černá
 - Bílá
 - Šedá
- Barva obrázku. Jsou podporovány následující barvy. Nejsou podporovány žádné další barvy.
 - Žlutá
 - Modrá
 - Červená
 - Fuchsiová (magenta)
 - Zelená
 - Modrá (cyan)
 - Předvolba (měla by se zobrazovat jako černá)
- Zoom.

Poznámka: Barva vybrané oblasti se nepoužívá pro webový prohlížeč AFP. Vybraná oblast se vždy zobrazuje s bílým textem a černým pozadím.

Image Web Viewer

Následující informace platí při používání webového prohlížeče obrázků k zobrazení vícestránkových obrázků.

Poznámka: Následující procedura vyžaduje, abyste upravili registr na počítači. Registr byste neměli upravovat, pokud to není absolutně nutné. Jestliže nastane chyba v registru, počítač nebude pracovat správně. Než budete pokračovat, měli byste vytvořit záložní kopii registru a dobře vědět, jak se registr obnovuje na stejnou verzi, kterou jste používali, když jste naposledy úspěšně spustili počítač. Návod najdete v informacích o systému Windows.

Když na vícestránkových obrázcích zatáhnete za svislý posuvník, vedle ouška se zobrazí malé okno. V tomto okně se zobrazuje počet stránek odpovídající poloze ouška a počtu stran v obrázku. Například 5 / 10 uvádí, že v obrázku je deset stran a že po uvolnění ouška bude aktuální strana číslo pět.

Toto chování může být potlačeno nastavením registru v tomto klíči:

HKEY_LOCAL_MACHINE\Software\IBM\OnDemand Image Web Viewer\Preferences

Jestliže hodnota řetězce `PageNumberScroll` je nastavena na 0 (nula), okno čísla strany se při tažení za ouško posuvníku nezobrazí.

Ve stejném klíči registru, pokud hodnota řetězce `PageNumberToolbar` je nastavena na 1 (jedna), se informace o počtu stran zobrazí na panelu nástrojů pro vícestránkové obrázky. Například 3 / 5 uvádí, že je pět stran v obrazu a momentálně je zobrazena strana číslo tři.

Java prohlížeč Line Data Viewer

IBM nyní poskytuje vylepšený Java prohlížeč Line Data Viewer. K funkčním vylepšením patří dokonalejší funkce tisku, například tisk na celou šířku strany. Grafické uživatelské rozhraní je založeno na knihovně Swing.

IBM nyní poskytuje dvě verze Java prohlížeče Line Data Viewer v adresáři appletů:

ODLineDataViewer.jar je starý Java prohlížeč Line Data Viewer, který vyžaduje u programu Java plugin verzi 1.1.8 nebo vyšší.

ODLineDataViewer2.jar je nový Java prohlížeč Line Data Viewer, který vyžaduje u programu Java plugin verzi 1.4.1 nebo vyšší.

Zákazníci mohou používat nový Java prohlížeč Line Data Viewer nebo starý Java prohlížeč Line Data Viewer. Výběr se uvádí nastavovacími parametry v části [DEFAULT BROWSER] v souboru ARSWWW.INI. Kromě toho, nový Java prohlížeč Line Data Viewer vyžaduje u programu Java plugin verzi 1.4.1 nebo vyšší. Další parametry v souboru ARSWWW.INI určují číslo verze a umístění instalačního souboru Java pro uživatele, kteří nemají na svých pracovních stanicích instalovanou požadovanou verzi programu Java plugin.

Tabulka 2 popisuje nové parametry v souboru ARSWWW.INI, které podporují Java prohlížeč Line Data Viewer.

Tabulka 2. Parametry v souboru ARSWWW.INI pro Java prohlížeč Line Data Viewer.

Parametr	Hodnota	Komentáře
ODApplet.version	1	Uvádí, že se má volat starý Java prohlížeč Line Data Viewer. Jestliže je zadán tento parametr, ignorujte zbývající parametry. Poznámka: Toto je předvolená hodnota. Navíc když je tento parametr vynechán, ODWEK použije starý Java prohlížeč Line Data Viewer.
	2	Uvádí, že se má volat nový Java prohlížeč Line Data Viewer (vylepšená verze). Jestliže je tento prohlížeč zadán, použijte následující tři parametry.

Tabulka 2. Parametry v souboru ARSWWW.INI pro Java prohlížeč Line Data Viewer. (pokračování)

Parametr	Hodnota	Komentáře
ODApplet.jre.path.IE	http://java.sun.com/getjava/installer.html	Pro Internet Explorer. Uvádí, že se má automaticky stahovat a instalovat poslední verze Java pluginu z webové stránky java.sun.com. Příklad toho, co se stane, když si uživatelé automaticky stahují a instalují Java plugin, najdete v dokumentu http://java.sun.com/getjava/install-windows.html . Poznámka: Uživatel bude možná muset po instalaci programu typu plugin restartovat prohlížeč.
	<location>	Uvádí umístění instalačního souboru programu Java pluginu ve firemní síti typu intranet. Umístění musí zahrnout platný protokol prohlížeče, například http, file nebo ftp. Příklad: file://shareName/java/plugins/plugin.exe Poznámka: Administrátor musí stáhnout instalační soubor Java pluginu a uložit jej na zadané místo. Po zadání umístění instalačního souboru prohlížeč automaticky nainstaluje Java plugin na pracovní stanici. Jakmile instalace skončí, uživatel bude možná muset restartovat prohlížeč.

Tabulka 2. Parametry v souboru ARSWWW.INI pro Java prohlížeč Line Data Viewer. (pokračování)

Parametr	Hodnota	Komentáře
ODApplet.jre.path.NN	http://java.sun.com/j2se/1.4.1/download.html	Pro Netscape. Uvádí, že se má otevřít stahovací stránka JRE/J2SE a na ní vybrat Java plugin, který se má instalovat. Uživatel bude postupovat podle odkazů a stáhne si instalační soubor Java pluginu pro svou platformu. Po stažení instalačního souboru Java pluginu musí uživatel nainstalovat plugin na pracovní stanici. Po instalaci pluginu bude uživatel možná muset restartovat prohlížeč.
	<location>	<p>Uvádí umístění souboru (souborů) pluginu ve firemní síti typu intranetu. Umístění musí zahrnovat platný protokol prohlížeče, například http, file nebo ftp. Příklad:</p> <pre>http://webServer/ tmp/ondemand/java/ plugins</pre> <p>Poznámka: Administrátor musí stáhnout soubor (soubory) pluginu a uložit ho na zadané místo. Nemůžete zadat cestu k určitému souboru, protože není známé, na kterém operačním systému je prohlížeč Netscape spuštěný. Zadaný formát také administrátorovi umožňuje v případě potřeby stáhnout plugin pro jiné platformy.</p> <p>Uživatel musí Java plugin nainstalovat na pracovní stanici. Po instalaci pluginu bude uživatel možná muset restartovat prohlížeč.</p>

Tabulka 2. Parametry v souboru ARSWWW.INI pro Java prohlížeč Line Data Viewer. (pokračování)

Parametr	Hodnota	Komentáře
ODApplet.jre.version	<version>	Uvádí verzi Java pluginu, která se má použít. Musí být uvedena verze 1.4 nebo pozdější. Zadejte číslo hlavní verze (například 1.4), která podporuje libovolné vydání programu typu plugin na této úrovni (například 1.4.0, 1.4.0_03, 1.4.1_01). Zadejte specifické číslo verze (například 1.4.1_01), která podporuje pouze tuto verzi Java pluginu. Získejte platná čísla verzí z webové stránky java.sun.com. Příklad: 1.4 nebo: 1.4.1_01

Následující příklad ukazuje, jak se konfiguruje soubor ARSWWW.INI pro podporu starého Java prohlížeče Line Data Viewer.

```
[DEFAULT BROWSER]
ODApplet.version=1
```

Poznámky:

1. Navíc když vynecháte parametr ODApplet.version v souboru ARSWWW.INI, ODWEK použije starý Java prohlížeč Line Data Viewer.
2. Parametry ODApplet mají globální rozsah a lze je zadat pouze v části DEFAULT BROWSER. (Kdyby tyto parametry byly zadány v některé jiné části prohlížeče, byly by ignorovány.)

Níže je uveden příklad, jak konfigurovat soubor ARSWWW.INI pro podporu nového Java prohlížeče Line Data Viewer (vylepšená verze) a verze 1.4 nebo vyšší verze Java pluginu. V případě prohlížeče Internet Explorer platí, že uživatelé mohou automaticky stahovat a instalovat poslední verze Java pluginu z webové stránky java.sun.com. V případě Netscape platí, že administrátor uložil kopie instalačních souborů Java pro různé platformy na zadané místo na lokálním webovém serveru, takže uživatelé nemusí přistupovat na stahovací stránku java.sun.com JRE/J2SE. **Poznámka:** Pouze ti uživatelé, kteří nemají na svých pracovních stanicích instalovanou verzi 1.4 nebo vyšší verzi Java pluginu budou vyzváni, aby si tento plugin stáhli a nainstalovali.

```
[DEFAULT BROWSER]
ODApplet.version=2
ODApplet.jre.path.IE=http://java.sun.com/getjava/installer.html
ODApplet.jre.path.NN=http://localWebServer/java/plugins
ODApplet.jre.version=1.4
```

Váš další krok

Jakmile nainstalujete software ODWEK, nakonfigurujete soubor ARSWWW.INI, nakonfigurujete vzorové aplikace a nainstalujete webové prohlížeče, můžete začít používat software ODWEK.

Dodatek A. Referenční informace pro rozhraní CGI API

Tato kapitola obsahuje informace o programovacích funkcích, které jsou k dispozici v aplikaci ODWEK. Tato kapitola by měla zajímat hlavně programátory odpovědné za integraci aplikace ODWEK do webových prohlížečů.

Poznámka: Hodnoty parametrů jsou standardní text. Je možné, že text bude obsahovat znaky, které zmatou webové prohlížeče. Aby se tedy předešlo možným chybám, musíte kódovat všechny speciální znaky jejich odpovídajícími hexadecimálními kódy. Tyto speciální znaky jsou například řídicí znaky a určité alfanumerické symboly. Například řetězec:

The post date is 12/31/95

bude převeden jako:

The%20post%20date%20is%2012%2f31%2f95

Khodnotám parametrů patří jména složek, jména polí složky a vyhledávací kritéria.

Funkce Add Annotation (Přidání anotace)

Přidání anotace k zadanému dokumentu.

Účel

Funkce Add Annotation (Přidání anotace) umožňuje uživatelům přidávat anotaci k zadanému dokumentu. K přidávání anotací musí mít uživatel oprávnění Add Annotation ve skupině aplikací OnDemand. (Oprávnění Access také umožňuje uživatelům přidávat anotace.)

Parametry

Tabulka 3. Funkce Add Annotation

Jméno=hodnota	Účel
<code>_function=addnote</code>	Přidání anotace
<code>_server=hodnota</code>	Jméno serveru OnDemand.
<code>_user=hodnota</code>	ID uživatele OnDemand. Uživatel musí mít oprávnění Add Annotation pro každou skupinu aplikací, která obsahuje dokumenty, jež mají být anotovány. (Oprávnění Application Group Access umožňuje uživatelům přidávat anotace.)
<code>_password=hodnota</code>	Heslo pro uživatele.
<code>_folder=hodnota</code>	Jméno složky.
<code>_perm=hodnota</code>	Určuje, zda anotace je Public (veřejná, 0), Private (soukromá, 1) nebo Private for Group (soukromá pro skupinu, 2). Veřejné anotace může prohlížet každý uživatel s oprávněním View Annotation pro danou skupinu aplikací. Soukromé anotace si může prohlížet uživatel, který anotaci vytvořil, administrátoři skupiny aplikací a administrátoři systému. Anotace soukromé pro skupinu mohou prohlížet uživatelé a systémoví administrátoři. Parametr <code>_group</code> obsahuje jméno skupiny. Předvolená hodnota je 0 (Public = veřejná).
<code>_group=groupName</code>	Jestliže je parametr <code>_perm</code> nastaven na 2 (Private for Group = soukromý pro skupinu), uvádí jméno skupiny.
<code>_copy=hodnota</code>	Určuje, zda by anotace měla zůstat připojena k dokumentu, když se dokument exportuje do jiného serveru. Předvolená hodnota je Off, což znamená, že anotace není připojena k dokumentu. Hodnota On znamená, že anotace je připojena k dokumentu, když se dokument exportuje do jiného serveru.
<code>_text=hodnota</code>	Text anotace

Tabulka 3. Funkce Add Annotation (pokračování)

Jméno=hodnota	Účel
<code>_html=hodnota</code>	<p>Určuje soubor HTML, který aplikace ODWEK používá jako šablonu pro generování výstupní webové stránky. Hodnota může být jméno souboru nebo * (hvězdička). Jestliže hodnota je hvězdička, aplikace ODWEK používá soubor ADDNOTE.HTML nalezený v adresáři zadaném v parametru <code>TEMPLATEDIR</code> v souboru <code>ARSWWW.INI</code>. Jestliže hodnota je jméno souboru beze jména cesty, soubor se musí nacházet v adresáři zadaném v parametru <code>TEMPLATEDIR</code>. Jestliže hodnota zahrnuje jméno cesty, cesta by se měla vztahovat k adresáři zadanému v parametru <code>TEMPLATEDIR</code>.</p> <p>Souhrnný obsah souboru HTML definuje zákazník. Soubor však musí obsahovat následující řádek komentáře:</p> <pre><!-- - -AOI# Marker-- - -></pre> <p>Umístění řádku komentáře určuje, kam aplikace ODWEK umísťuje svůj výstup. Všechny řádky uvedené nad řádkem komentáře se zapíší před výstup generovaný aplikací ODWEK. Všechny řádky uvedené pod řádkem komentáře se zapíší za výstup generovaný aplikací ODWEK.</p> <p>Soubor <code>TEMPLATE.HTM</code> je vzorový soubor šablony dodávaný aplikací ODWEK. Pomocí vzorového souboru šablony můžete vytvořit vlastní soubor šablony pro funkci přidání anotace.</p>
<code>_nohtml=hodnota</code>	<p>Určuje typ výstupu generovaného aplikací ODWEK. Předvolená hodnota je 0 (nula) a znamená, že aplikace ODWEK generuje výstup HTML. Jestliže zadáte hodnotu 1 (jedna), aplikace ODWEK generuje výstup ASCII s oddělovači. Podrobnosti o výstupu ASCII s oddělovači najdete v tématu Dodatek H, "Žádný HTML výstup", na stránce 131.</p>
<code>_docid=documentID</code>	<p>Identifikátor dokumentu, k němuž se anotace má připojit. Identifikátor dokumentů vrácený funkcí Document Hit List.</p>
<code>_port=hodnota</code>	<p>Číslo portu pro server OnDemand. Předvolená hodnota 0 (nula) znamená, že server používá číslo portu, které je uvedeno v tabulce služeb (<code>WRKSRVTBLE</code>). Jestliže v tabulce služeb není uvedeno číslo portu, server OnDemand se pokusí použít číslo portu 1445. Každá vámi zadaná hodnota potlačí hodnotu parametru <code>PORT</code> v souboru <code>ARSWWW.INI</code>.</p>
<code>_codepage=hodnota</code>	<p>Kódová stránka databáze OnDemand. Předvolená kódová stránka je kódová stránka HTTP serveru. Pokud kódová stránka serveru je jiná než kódová stránka databáze, musíte kódovou stránku zadat. Každá vámi zadaná hodnota potlačí hodnotu parametru <code>CODEPAGE</code> v souboru <code>ARSWWW.INI</code>.</p>
<code>_logoff=1</code>	<p>Automaticky odpojuje uživatele od serveru OnDemand po přidání anotace. Když zadáte tento parametr, není zapotřebí, aby vaše aplikace volala funkci Logoff pro odpojení uživatele. Jediná platná hodnota pro tento parametr je 1 (jedna).</p>

Užití

Tyto parametry jsou povinné:

```
_function
_server
_user
_password
_text
_docid
```

Tyto parametry jsou volitelné:

```
_perm
_group (požaduje se, pokud _perm uvádí Private for Group)
```

| _html
| _nohtml
| _port
| _codepage
| _logoff

Sample Function Call

| http://www.company.com/cgi-bin/arswww.cgi?_function=addnote
| &_server=od400&_user=web&_password=web
| &_folder=credit%20card%20statements
| &_text=Test%20note%20from%20the%20OnDemand%20Internet%20Client
| &_docid=6850-6851-SUA17-1FAAA-225712-1634-132014-132172-89-76-11-25-0
| &_perm=1&_logoff=1

Funkce Change Password (Změna hesla)

Změna přihlašovacího hesla OnDemand.

Účel

Funkce Change Password (Změna hesla) umožňuje uživatelům měnit jejich hesla serveru OnDemand.

Parametry

Tabulka 4. Funkce Změna hesla

Jméno=hodnota	Účel
_function=chgpasword	Mění heslo OnDemand pro ID uživatele.
_server=hodnota	Jméno serveru OnDemand.
_user=hodnota	ID uživatele OnDemand.
_password=hodnota	Heslo pro ID uživatele.
_new_password=hodnota	Nové heslo pro ID uživatele.
_html=hodnota	<p>Určuje soubor HTML, který aplikace ODWEK používá jako šablonu pro generování výstupní webové stránky. Hodnota může být jméno souboru nebo * (hvězdička). Jestliže hodnota je hvězdička, pak aplikace ODWEK používá soubor CHGPASSWORD.HTML umístěný v adresáři zadaném v parametru TEMPLATEDIR v souboru ARSWWW.INI. Jestliže hodnota je jméno souboru beze jména cesty, soubor se musí nacházet v adresáři zadaném v parametru TEMPLATEDIR. Jestliže hodnota zahrnuje jméno cesty, cesta by se měla vztahovat k adresáři zadanému v parametru TEMPLATEDIR.</p> <p>Souhrnný obsah souboru HTML definuje zákazník. Soubor však musí obsahovat následující řádek komentáře:</p> <pre><!-- -AOI# Marker-- -></pre> <p>Umístění řádku komentáře určuje, kam aplikace ODWEK umísťuje svůj výstup. Všechny řádky uvedené nad řádkem komentáře se zapíší před výstup generovaný aplikací ODWEK. Všechny řádky uvedené pod řádkem komentáře se zapíší za výstup generovaný aplikací ODWEK.</p> <p>Soubor TEMPLATE.HTM je vzorový soubor šablony dodávaný aplikací ODWEK. Pomocí vzorového souboru šablony můžete vytvořit vlastní soubor šablony pro funkci změny hesla.</p>
_nohtml=hodnota	Určuje typ výstupu generovaného aplikací ODWEK. Předvolená hodnota je 0 (nula) a znamená, že aplikace ODWEK generuje výstup HTML. Jestliže zadáte hodnotu 1 (jedna), aplikace ODWEK generuje výstup ASCII s oddělovači. Podrobnosti o výstupu ASCII s oddělovači najdete v tématu Dodatek H, "Žádný HTML výstup", na stránce 131.
_port=hodnota	Číslo portu pro server OnDemand. Předvolená hodnota 0 (nula) znamená, že server používá číslo portu, které je uvedeno v tabulce služeb (WRKSRVTBLE). Jestliže v tabulce služeb není uvedeno číslo portu, server OnDemand se pokusí použít číslo portu 1445. Každá vámi zadaná hodnota potlačí hodnotu parametru PORT v souboru ARSWWW.INI.
_codepage=hodnota	Kódová stránka databáze OnDemand. Předvolená kódová stránka je kódová stránka HTTP serveru. Pokud kódová stránka serveru je jiná než kódová stránka databáze, musíte kódovou stránku zadat. Každá vámi zadaná hodnota potlačí hodnotu parametru CODEPAGE v souboru ARSWWW.INI.

Tabulka 4. Funkce Změna hesla (pokračování)

Jméno=hodnota	Účel
<code>_cgibin=program</code>	<p>Používá se programem CGI, když se generuje další výstupní strana. Pokud je zadáno, stránka bude obsahovat volání zadaného programu namísto předvoleného programu (ARSWWW.CGI). Tento parametr používají především programátoři, kteří vytvářejí přední program CGI nebo servlet pro program CGI nebo servlet dodávaný IBM.</p> <p>Parametr <i>program</i> může uvádět adresář, který se vztahuje k direktivě <code>ServerRoot</code> nebo jménu <i>alias</i>, které je definováno v konfiguračním souboru HTTP serveru. Aplikace ODWEK standardně načítá program CGI z adresáře CGI-BIN.</p>
<code>_logoff=1</code>	<p>Po změně hesla automaticky odpojuje uživatele od serveru OnDemand. Když zadáte tento parametr, není zapotřebí, aby vaše aplikace volala funkci <code>Logoff</code> pro odpojení uživatele. Jediná platná hodnota pro tento parametr je 1 (jedna).</p>

Užití

Tyto parametry jsou povinné:

- `_function`
- `_server`
- `_user`
- `_password`
- `_new_password`

Tyto parametry jsou volitelné:

- `_html`
- `_nohtml`
- `_port`
- `_codepage`
- `_logoff`
- `_cgibin`

Sample Function Call

```
http://www.company.com/cgi-bin/arswww.cgi?_function=chgpassword
&_server=od400&_user=web&_password=web
&_newpassword=newpw&_html=template.htm&_logoff=1
```

Funkce Document Hit List (Seznam nalezených dokumentů)

Zobrazuje seznam dokumentů, které odpovídají vyhledávacím kritériím

Účel

Funkce Document Hit List (Seznam nalezených dokumentů) zobrazuje seznam dokumentů, které odpovídají vyhledávacím kritériím pro určitou složku. Každý dokument je uváděn jako odkaz na dokument na serveru OnDemand. Po klepnutí na dokument aplikace ODWEK načte ze serveru dokument a zobrazí jej v okně webového prohlížeče pomocí příslušného prohlížeče.

Parametry

Tabulka 5. funkce Document Hit List

Jméno=hodnota	Účel
_function=dochitlist	Zobrazuje seznam dokumentů, které odpovídají vyhledávacím kritériím.
_server=hodnota	Jméno serveru OnDemand.
_user=hodnota	ID uživatele OnDemand.
_password=hodnota	Heslo pro ID uživatele.
_folder=hodnota	Jméno složky.
<i>jméno pole složky=hodnota</i>	Jméno vyhledávacího pole složky a vyhledávací hodnota. Můžete zadat jednu nebo více sad jmen polí a vyhledávací hodnoty, a to až do počtu polí definovaných pro složku.
<i>jméno pole složky2=hodnota</i>	Pro vyhledávací pole složky, která používají vyhledávací operátory BETWEEN nebo NOT BETWEEN, je to horní hodnota, se kterou se prohledává pole.
<i>jméno pole složkyOP=hodnota</i>	Operátor, který se má použít pro potlačení výchozího operátoru pro vyhledávací pole složky. <i>Hodnota</i> musí být jedna z následujících: 1 znamená Equal (Rovno) 2 znamená Not Equal (Nerovno) 4 znamená Less Than (Menší než) 8 znamená Less Than or Equal (Menší než nebo rovno) 16 znamená Greater Than (Větší než) 32 znamená Greater Than or Equal (Větší než nebo rovno) 64 znamená In (V) 128 znamená Not In (Ne v) 256 znamená Like (Podobné) 512 znamená Not Like (Nepodobné) 1024 znamená Between (Mezi) 2048 znamená Not Between (Ne mezi)
_zobrazovaná_pole=hodnota[,hodnota,...]	Seznam dělený čárkami obsahující jména zobrazovaných polí složky. Můžete zadat jedno nebo více jmen polí složky. Pokud nezadáte tento parametr, výstupní stránka bude obsahovat všechna zobrazovaná pole složky.
_třídící_pole=hodnota[,hodnota,...]	Určuje vyhledávací pole složky, které server OnDemand používá pro třídění položek v seznamu dokumentů. Jestliže zadáte více než jedno pole, jména polí oddělíte čárkou. Příklad: _třídící_pole=Účet,Účet+Zůstatek,Datum . Předvolená třídící pole jsou definována na straně Field Information pro složku.
_třídící_pořadí=hodnota[,hodnota,...]	Pro každé vyhledávací pole složky, které je zadáno v parametru sort_field , určuje, zda server OnDemand třídí položky od první po poslední nebo od poslední po první. Chcete-li seznam třídít od první po poslední, zadejte A . Chcete-li třídít položky od poslední po první, zadejte libovolný jiný znak. Například: _sort_order=A,D,A . Předvolené třídící pořadí se určuje podle třídícího pořadí, které je definováno na straně Field Information pro danou složku.

Tabulka 5. funkce Document Hit List (pokračování)

Jméno=hodnota	Účel
_max_hits=hodnota	<p>Určuje maximální počet položek, které aplikace ODWEK vrací do seznamu dokumentů, nehledě na počet položek, které odpovídají dotazu. Aplikace ODWEK vyplní seznam dokumentů položkami, které odpovídají dotazu, a to v pořadí, v jakém byly odpovídající položky zavedeny do databáze.</p> <p>Aplikace ODWEK používá první zadanou hodnotu ke zjištění počtu položek, který se vrací do seznamu dokumentů:</p> <ol style="list-style-type: none"> 1. Hodnota pole Maximum Hits (zadaná na straně Permissions příslušné složky). Tato hodnota potlačí všechny jiné hodnoty. 2. Hodnota parametru _max_hits, pokud je zadána. Tato hodnota potlačí parametr MAXHITS ze souboru ARSWWW.INI. 3. Hodnota parametru MAXHITS, pokud je zadána. 4. Jestliže není zadána žádná z výše uvedených hodnot, aplikace ODWEK vrátí do seznamu dokumentů maximálně 200 položek.
_html=hodnota	<p>Určuje soubor HTML, který aplikace ODWEK používá jako šablonu pro generování výstupní webové stránky. Hodnota může být jméno souboru nebo * (hvězdička). Jestliže hodnota je hvězdička, pak aplikace ODWEK používá soubor DOCHITLIST.HTML umístěný v adresáři zadaném v parametru TEMPLATEDIR v souboru ARSWWW.INI. Jestliže hodnota je jméno souboru beze jména cesty, soubor se musí nacházet v adresáři zadaném v parametru TEMPLATEDIR. Jestliže hodnota zahrnuje jméno cesty, pak by se cesta měla vztahovat k adresáři zadanému v parametru TEMPLATEDIR.</p> <p>Souhrnný obsah souboru HTML definuje zákazník. Soubor však musí obsahovat následující řádek komentáře:</p> <pre><!-- -AOI# Marker-- -></pre> <p>Umístění řádku komentáře určuje, kam aplikace ODWEK umísťuje svůj výstup. Všechny řádky uvedené nad řádkem komentáře se zapíší před výstup generovaný aplikací ODWEK. Všechny řádky uvedené pod řádkem komentáře se zapíší za výstup generovaný aplikací ODWEK.</p> <p>Soubor TEMPLATE.HTM je vzorový soubor šablony dodávaný aplikací ODWEK. Pomocí vzorového souboru šablony můžete vytvořit vlastní soubor šablony pro funkce nalezených výskytů v dokumentech.</p>
_frame=hodnota	<p>Výstup tohoto příkazu bude zahrnovat atribut target=hodnota. Tento parametr usnadňuje vytváření rámců HTML. Jedná se o volitelný parametr.</p>
_datefmt=hodnota	<p>Určuje formát hodnoty data používaný aplikací ODWEK k zobrazení položek, které odpovídají kritériím dotazu. Předvolený formát data je nastaven na straně Field Information příslušné složky. Podrobnosti o formátech podporovaných na serveru OnDemand najdete v publikaci <i>IBM Content Manager OnDemand for iSeries V5R1 Common Server Administration Guide</i>, SC27-1161.</p>
_nohtml=hodnota	<p>Určuje typ výstupu generovaného aplikací ODWEK. Předvolená hodnota je 0 (nula) a znamená, že aplikace ODWEK generuje výstup HTML. Jestliže zadáte hodnotu 1 (jedna), aplikace ODWEK generuje výstup ASCII s oddělovači. Podrobnosti o výstupu ASCII s oddělovači najdete v tématu Dodatek H, “Žádný HTML výstup”, na stránce 131.</p>
_port=hodnota	<p>Číslo portu pro server OnDemand. Předvolená hodnota 0 (nula) znamená, že server používá číslo portu, které je uvedeno v tabulce služeb (WRKSRVTBLE). Jestliže v tabulce služeb není uvedeno číslo portu, server OnDemand se pokusí použít číslo portu 1445. Každá vami zadaná hodnota potlačí hodnotu parametru PORT v souboru ARSWWW.INI.</p>

Tabulka 5. funkce Document Hit List (pokračování)

Jméno=hodnota	Účel
<code>_codepage=hodnota</code>	Kódová stránka databáze OnDemand. Předvolená kódová stránka je kódová stránka HTTP serveru. Pokud kódová stránka serveru je jiná než kódová stránka databáze, musíte kódovou stránku zadat. Každá vámi zadaná hodnota potlačí hodnotu parametru CODEPAGE v souboru ARSWWW.INI.
<code>_sql=řetězec</code>	Uvádí dotaz SQL, který server OnDemand používá pro prohledání složky. Pokud zadáte tuto hodnotu, dotaz SQL se použije k prohledání složky namísto jiné dvojice jméno/hodnota složky, která je snad zadána. Server OnDemand neověřuje řetězec dotazu. Když používáte řetězec SQL, musíte zadat jména a hodnoty databázových polí skupiny aplikací. Jestliže plánujete dotazy na datumová pole, musíte zadat interní datumové hodnoty serveru OnDemand. Například datum 1. ledna 1999 se zadává jako 10593. Můžete použít příkaz ARSDATE pro výpis interních hodnot pro příslušné datum. Řetězec SQL se používá k prohledávání všech skupin aplikací, které jsou obsaženy ve složce. Jestliže řetězec SQL obsahuje jméno databázového pole, které je v jedné skupině aplikací, ale ne v jiné skupině aplikací, dotaz selže.
<code>_date1=hodnota</code>	Používá se pro zadání počátečního data v rozsahu prohledávaných dat. Jestliže zadáte parametry <code>_date1</code> a <code>_date2</code> , server OnDemand omezí dotaz na tabulky nebo tabulky, které obsahují jedno nebo obě zadaná data. Formát datumového řetězce, který zadáte, musí odpovídat formátu zobrazení pole složky. (K zobrazení formátu pole složky můžete použít administrativního klienta.)
<code>_date2=hodnota</code>	Používá se pro zadání konečného data v rozsahu prohledávaných dat. Jestliže zadáte parametry <code>_date1</code> a <code>_date2</code> , server OnDemand omezí dotaz na tabulky nebo tabulky, které obsahují jedno nebo obě zadaná data. Formát datumového řetězce, který zadáte, musí odpovídat formátu zobrazení pole složky. (K zobrazení formátu pole složky můžete použít administrativního klienta.)
<code>_cgibin=program</code>	Používá se programem CGI, když se generuje další výstupní strana. Pokud je zadáno, stránka bude obsahovat volání zadaného programu namísto předvoleného programu (ARSWWW.CGI). Tento parametr používají především programátoři, kteří vytvářejí přední program CGI nebo servlet pro program CGI nebo servlet dodávaný IBM. Parametr <i>program</i> může uvádět adresář, který se vztahuje k direktivě <code>ServerRoot</code> nebo jménu <i>alias</i> , které je definováno v konfiguračním souboru HTTP serveru. Aplikace ODWEK standardně načítá program CGI z adresáře <code>/QIBM/Proddata/OnDemand/www/bin</code> .
<code>_or=hodnota</code>	Chcete-li vyhledávací pole spojit logickým operátorem OR, zadejte 1 (jedna); položka musí odpovídat alespoň jedné ze zadaných vyhledávacích hodnot. Předvolená hodnota je 0 (nula), což znamená, že server OnDemand spojuje vyhledávací pole logickým operátorem AND (položka musí odpovídat všem zadaným vyhledávacím hodnotám).
<code>_logoff=1</code>	Po vytvoření seznamu dokumentů automaticky odpojuje uživatele od serveru OnDemand. Když zadáte tento parametr, není zapotřebí, aby vaše aplikace volala funkci <code>Logoff</code> pro odpojení uživatele. Jediná platná hodnota pro tento parametr je 1 (jedna).

Užití

Tyto parametry jsou povinné:

- `_function`
- `_server`
- `_user`
- `_password`
- `_folder`

Tyto parametry jsou volitelné:

jméno pole složky
jméno pole složky2
jméno pole složkyOP
_display_fields
_sort_field
_sort_order
_max_hits
_frame
_datefmt
_sql
_date1
_date2
_or
_html
_nohtml
_port
_codepage
_logoff
_cgibin

Sample Function Call

```
http://www.company.com/cgi-bin/arswww.cgi?_function=dochitlist
&_server=od400&_user=web&_password=web
&_folder=credit%20card%20statements
&account%20number=1000100010009999&date=1%2f1%2f96&date2=12%2f31%2f96
&nameOP=256&name=%AA
&_sort_field=Account,Account%20Balance,Date&_sort_order=A,D,A
&_logoff=1
&_html=template.htm
```


Funkce Logoff (Odhlášení)

Odhlášení od serveru OnDemand.

Účel

Funkce Logoff (Odhlášení) slouží k odhlašování uživatele od serveru OnDemand. Jméno serveru a ID uživatele k odhlášení se ukládají v cookie browser na klientovi pomocí funkce Logon (Odhlášení). Jestliže server není platný server OnDemand, vrátí se chybová zpráva. Jestliže ID uživatele není přihlášen k zadanému serveru, vrátí se chybová zpráva.

Parametry

Tabulka 6. funkce Logoff

Jméno=hodnota	Účel
<code>_function=logoff</code>	Odhlášení od serveru OnDemand.
<code>_html=hodnota</code>	<p>Určuje soubor HTML, který aplikace ODWEK používá jako šablonu pro generování výstupní webové stránky. Hodnota může být jméno souboru nebo * (hvězdička). Jestliže hodnota je hvězdička, pak aplikace ODWEK používá soubor LOGOFF.HTML umístěný v adresáři zadaném v parametru <code>TEMPLATEDIR</code> v souboru <code>ARSWWW.INI</code>. Jestliže hodnota je jméno souboru bez jména cesty, soubor se musí nacházet v adresáři zadaném v parametru <code>TEMPLATEDIR</code>. Jestliže hodnota zahrnuje jméno cesty, pak by se cesta měla vztahovat k adresáři zadanému v parametru <code>TEMPLATEDIR</code>.</p> <p>Souhrnný obsah souboru HTML definuje zákazník. Soubor však musí obsahovat následující řádek komentáře:</p> <pre><!-- -AOI# Marker-- -></pre> <p>Umístění řádku komentáře určuje, kam aplikace ODWEK umísťuje svůj výstup. Všechny řádky uvedené nad řádkem komentáře se zapíší před výstup generovaný aplikací ODWEK. Všechny řádky uvedené pod řádkem komentáře se zapíší za výstup generovaný aplikací ODWEK.</p> <p>Soubor <code>TEMPLATE.HTM</code> je vzorový soubor šablony dodávaný aplikací ODWEK. Pomocí vzorového souboru šablony můžete vytvořit vlastní soubor šablony pro funkci odhlášení.</p>
<code>_nohtml=hodnota</code>	Určuje typ výstupu generovaného aplikací ODWEK. Předvolená hodnota je 0 (nula) a znamená, že aplikace ODWEK generuje výstup HTML. Jestliže zadáte hodnotu 1 (jedna), aplikace ODWEK generuje výstup ASCII s oddělovači. Podrobnosti o výstupu ASCII s oddělovači najdete v tématu Dodatek H, "Žádný HTML výstup", na stránce 131.
<code>_port=hodnota</code>	Číslo portu pro server OnDemand. Předvolená hodnota 0 (nula) znamená, že server používá číslo portu, které je uvedeno v tabulce služeb (<code>WRKSRVTBLE</code>). Jestliže v tabulce služeb není uvedeno číslo portu, server OnDemand se pokusí použít číslo portu 1445. Každá vámi zadaná hodnota potlačí hodnotu parametru <code>PORT</code> v souboru <code>ARSWWW.INI</code> .

Užití

Tyto parametry jsou povinné:

`_function`

Tyto parametry jsou volitelné:

`_html`

`_nohtml`

`_port`

Sample Function Call

```
http://www.company.com/cgi-bin/arswww.cgi?_function=logoff  
&_html=template.htm
```

Funkce Logon (Přihlášení)

Přihlášení k serveru OnDemand

Účel

Funkce Logon (Přihlášení) slouží pro přístup k serveru OnDemand pomocí hodnot parametrů "server", "user" a "password". Funkce Logon (Přihlášení) ověřuje, zda zadaný uživatel je oprávněn přihlašovat se k zadanému serveru, a jeho heslo. Jestliže uživatel není oprávněn přihlašovat se k serveru, vrátí se chybová zpráva. Jestliže server není platný server OnDemand, vrátí se chybová zpráva. Jestliže heslo není pro uživatele platné, vrátí se chybová zpráva. Po úspěšném přihlášení funkce Logon zobrazí webovou stránku, která obsahuje seznam složek, k nimž je uživatel oprávněn přistupovat.

Parametry

Tabulka 7. funkce Logon

Jméno=hodnota	Účel
<code>_function=logon</code>	Přihlášení k serveru OnDemand.
<code>_server=hodnota</code>	Jméno serveru OnDemand.
<code>_user=hodnota</code>	ID uživatele OnDemand.
<code>_password=hodnota</code>	Heslo pro ID uživatele.
<code>_new_password=hodnota</code>	Nové heslo pro ID uživatele. Umožňuje změnit heslo po úspěšném přihlášení k serveru OnDemand. Jedná se o volitelný parametr.
<code>_html=hodnota</code>	<p>Určuje soubor HTML, který aplikace ODWEK používá jako šablonu pro generování výstupní webové stránky. Hodnota může být jméno souboru nebo * (hvězdička). Jestliže hodnota je hvězdička, pak aplikace ODWEK používá soubor LOGON.HTML umístěný v adresáři zadaném v parametru <code>TEMPLATEDIR</code> v souboru <code>ARSWWW.INI</code>. Jestliže hodnota je jméno souboru beze jména cesty, soubor se musí nacházet v adresáři zadaném v parametru <code>TEMPLATEDIR</code>. Jestliže hodnota zahrnuje jméno cesty, pak by se cesta měla vztahovat k adresáři zadanému v parametru <code>TEMPLATEDIR</code>.</p> <p>Souhrnný obsah souboru HTML definuje zákazník. Soubor však musí obsahovat následující řádek komentáře:</p> <pre><!-- -AOI# Marker- -></pre> <p>Umístění řádku komentáře určuje, kam aplikace ODWEK umísťuje svůj výstup. Všechny řádky uvedené nad řádkem komentáře se zapíší před výstup generovaný aplikací ODWEK. Všechny řádky uvedené pod řádkem komentáře se zapíší za výstup generovaný aplikací ODWEK.</p> <p>Soubor <code>TEMPLATE.HTM</code> je vzorový soubor šablony dodávaný aplikací ODWEK. Pomocí vzorového souboru šablony můžete vytvořit vlastní soubor šablony pro funkci přihlášení.</p>
<code>_frame=hodnota</code>	Výstup tohoto příkazu bude zahrnovat atribut <code>target=hodnota</code> . Tento parametr usnadňuje vytváření rámců HTML. Jedná se o volitelný parametr.
<code>_datefmt=hodnota</code>	Určuje formát hodnoty data používaný aplikací ODWEK k zobrazení položek, které odpovídají kritériím dotazu. Předvolený formát data je nastaven na straně Field Information příslušné složky. Podrobnosti o formátech podporovaných na serveru OnDemand najdete v publikaci <i>IBM Content Manager OnDemand for iSeries V5R1 Common Server Administration Guide</i> , SC27-1161 .

Tabulka 7. funkce Logon (pokračování)

Jméno=hodnota	Účel
<code>_nohtml=hodnota</code>	Určuje typ výstupu generovaného aplikací ODWEK. Předvolená hodnota je 0 (nula) a znamená, že aplikace ODWEK generuje výstup HTML. Jestliže zadáte hodnotu 1 (jedna), aplikace ODWEK generuje výstup ASCII s oddělovači. Podrobnosti o výstupu ASCII s oddělovači najdete v tématu Dodatek H, “Žádný HTML výstup”, na stránce 131.
<code>_port=hodnota</code>	Číslo portu pro server OnDemand. Předvolená hodnota 0 (nula) znamená, že server používá číslo portu, které je uvedeno v tabulce služeb (WRKSRVTBLE). Jestliže v tabulce služeb není uvedeno číslo portu, server OnDemand se pokusí použít číslo portu 1445. Každá vámi zadaná hodnota potlačí hodnotu parametru PORT v souboru ARSWWW.INI.
<code>_codepage=hodnota</code>	Kódová stránka databáze OnDemand. Předvolená kódová stránka je kódová stránka HTTP serveru. Pokud kódová stránka serveru je jiná než kódová stránka databáze, musíte kódovou stránku zadat. Každá vámi zadaná hodnota potlačí hodnotu parametru CODEPAGE v souboru ARSWWW.INI.
<code>_cgibin=program</code>	<p>Používá se programem CGI, když se generuje další výstupní strana. Pokud je zadáno, stránka bude obsahovat volání zadaného programu namísto předvoleného programu (ARSWWW.CGI). Tento parametr používají především programátoři, kteří vytvářejí přední program CGI nebo servlet pro program CGI nebo servlet dodávaný IBM.</p> <p>Parametr <i>program</i> může uvádět adresář, který se vztahuje k direktivě ServerRoot nebo jménu <i>alias</i>, které je definováno v konfiguračním souboru HTTP serveru. Aplikace ODWEK standardně načítá program CGI z adresáře CGI-BIN.</p>

Užití

Tyto parametry jsou povinné:

```
_function
_server
_user
_password
```

Tyto parametry jsou volitelné:

```
_new_password
_frame
_datefmt
_html
_nohtml
_port
_codepage
_logoff
_cgibin
```

Sample Function Call

```
http://www.company.com/cgi-bin/arswww.cgi?_function=logon
&_server=od400&_user=web&_password=web
&_html=template.htm
```

Funkce Print Document (Server) (Tisk dokumentu)

Odesílá jeden nebo více dokumentů na zadanou serverovou tiskárnu.

Účel

Funkce Print Document (Tisk dokumentu) odesílá kopie dokumentů na tiskárnu serveru OnDemand. K používání serverového tisku musí mít uživatel oprávnění Print Document ve skupině aplikací OnDemand. (Oprávnění Access také umožňuje uživatelům tisknout dokumenty.) Na zadaném serveru OnDemand musí být definována alespoň jedna serverová tiskárna.

Parametry

Tabulka 8. Funkce Print Document

Jméno=hodnota	Účel
_function=printdocs	Tiskne dokumenty.
_server=hodnota	Jméno serveru OnDemand.
_user=hodnota	ID uživatele OnDemand. Uživatel musí mít oprávnění Document Print pro každou skupinu aplikací, která obsahuje dokumenty, jež mají být tištěny. (Oprávnění Application Group Access umožňuje uživatelům tisknout dokumenty.)
_password=hodnota	Heslo pro uživatele.
_folder=hodnota	Jméno složky.
_printer=hodnota	<p>Jméno serverové tiskárny OnDemand.</p> <p>Pokud zadaná tiskárna je FAX nebo Printer with Information, pak můžete zadat následující dodatečné parametry:</p> <p>_recv_name=hodnota Jméno příjemce.</p> <p>_recv_comp=hodnota Jméno společnosti/firmy příjemce.</p> <p>_recv_fax=hodnota Číslo faxu příjemce.</p> <p>_send_name=hodnota Jméno odesílatele.</p> <p>_send_comp=hodnota Jméno společnosti/firmy odesílatele.</p> <p>_send_tel=hodnota Telefonní číslo odesílatele.</p> <p>_send_fax=hodnota Faxové číslo odesílatele.</p> <p>_send_cover=hodnota Uživatelsky definovaný překryt, který program Header Page Exit spojuje s hodnotami jiných parametrů pro vytvoření titulní strany dokumentu.</p> <p>_subject=hodnota Řetězec, který představuje téma dokumentu.</p> <p>_notes=hodnota Řetězec, který představuje poznámku o dokumentu.</p>

Tabulka 8. Funkce Print Document (pokračování)

Jméno=hodnota	Účel
<p>_html=hodnota</p>	<p>Určuje soubor HTML, který aplikace ODWEK používá jako šablonu pro generování výstupní webové stránky. Hodnota může být jméno souboru nebo * (hvězdička). Jestliže hodnota je hvězdička, pak aplikace ODWEK používá soubor PRINTDOCS.HTML umístěný v adresáři zadaném v parametru TEMPLATEDIR v souboru ARSWWW.INI. Jestliže hodnota je jméno souboru bez jména cesty, soubor se musí nacházet v adresáři zadaném v parametru TEMPLATEDIR. Jestliže hodnota zahrnuje jméno cesty, pak by se cesta měla vztahovat k adresáři zadanému v parametru TEMPLATEDIR.</p> <p>Souhrnný obsah souboru HTML definuje zákazník. Soubor však musí obsahovat následující řádek komentáře:</p> <pre><!-- - -AOI# Marker-- - -></pre> <p>Umístění řádku komentáře určuje, kam aplikace ODWEK umísťuje svůj výstup. Všechny řádky uvedené nad řádkem komentáře se zapíší před výstup generovaný aplikací ODWEK. Všechny řádky uvedené pod řádkem komentáře se zapíší za výstup generovaný aplikací ODWEK.</p> <p>Soubor TEMPLATE.HTM je vzorový soubor šablony dodávaný aplikací ODWEK. Pomocí vzorového souboru šablony můžete vytvořit vlastní soubor šablony pro funkci tisku dokumentů.</p>
<p>_nohtml=hodnota</p>	<p>Určuje typ výstupu generovaného aplikací ODWEK. Předvolená hodnota je 0 (nula) a znamená, že aplikace ODWEK generuje výstup HTML. Jestliže zadáte hodnotu 1 (jedna), aplikace ODWEK generuje výstup ASCII s oddělovači. Podrobnosti o výstupu ASCII s oddělovači najdete v tématu Dodatek H, “Žádný HTML výstup”, na stránce 131.</p>
<p>_docids=documentIDList</p>	<p>Seznam identifikátorů dokumentů pro tištěné dokumenty. Identifikátory dokumentů jsou vráceny funkcí Document Hit List. Jestliže zadáte více než jeden identifikátor dokumentu, pak musíte identifikátory dokumentu oddělit znakem \003.</p> <p>Poznámka: Jestliže počet identifikátorů znaků přesáhne 200, pak musíte zadat parametr _max_hits.</p>
<p>_port=hodnota</p>	<p>Číslo portu pro server OnDemand. Předvolená hodnota 0 (nula) znamená, že server používá číslo portu, které je uvedeno v tabulce služeb (WRKSRVTBLE). Jestliže v tabulce služeb není uvedeno číslo portu, server OnDemand se pokusí použít číslo portu 1445. Každá vámi zadaná hodnota potlačí hodnotu parametru PORT v souboru ARSWWW.INI.</p>
<p>_codepage=hodnota</p>	<p>Kódová stránka databáze OnDemand. Předvolená kódová stránka je kódová stránka HTTP serveru. Pokud kódová stránka serveru je jiná než kódová stránka databáze, musíte kódovou stránku zadat. Každá vámi zadaná hodnota potlačí hodnotu parametru CODEPAGE v souboru ARSWWW.INI.</p>

Tabulka 8. Funkce Print Document (pokračování)

Jméno=hodnota	Účel
<code>_max_hits=hodnota</code>	<p>Tento parametr použijte k zadání počtu identifikátorů dokumentů, které se mají zpracovat. Zadejte hodnotu, která je rovna nebo větší než počet identifikátorů zadaný parametrem <code>_docids</code>.</p> <p>Poznámka: Jestliže počet identifikátorů dokumentů přesáhne hodnotu zadanou parametrem <code>MAXHITS</code> v souboru <code>ARSWWW.CGI</code> (nebo 200, pokud hodnota není zadána), pak musíte zadat parametr <code>_max_hits</code>. Jestliže neuvedete parametr <code>_max_hits</code> (nebo neuvedete hodnotu pro parametr <code>MAXHITS</code>), zpracuje se maximálně 200 identifikátorů dokumentů, nehledě na počet identifikátorů dokumentů, který jste uvedli v parametru <code>_docids</code>.</p> <p>Aplikace ODWEK používá jednu z následujících hodnot k určení počtu identifikátorů dokumentů, které se zpracovávají:</p> <ul style="list-style-type: none"> • Hodnota parametru <code>_max_hits</code>, pokud je zadána. Tato hodnota potlačí hodnotu parametru <code>MAXHITS</code>. • Hodnota parametru <code>MAXHITS</code>, pokud je zadána. • Pokud není zadána žádná z výše uvedených hodnot, aplikace ODWEK zpracuje maximum 200 identifikátorů dokumentů.
<code>_logoff=1</code>	<p>Po vytištění dokumentu automaticky odpojí uživatele od serveru OnDemand. Když zadáte tento parametr, není zapotřebí, aby vaše aplikace volala funkci <code>Logoff</code> pro odpojení uživatele. Jediná platná hodnota pro tento parametr je 1 (jedna).</p>

Užití

Tyto parametry jsou povinné:

`_function`
`_server`
`_user`
`_password`
`_folder`
`_printer`
`_docids`

Tyto parametry jsou volitelné:

`_recv_name`
`_recv_comp`
`_recv_fax`
`_send_name`
`_send_comp`
`_send_tel`
`_send_fax`
`_send_cover`
`_subject`
`_notes`
`_max_hits`
`_html`
`_nohtml`
`_port`
`_codepage`
`_logoff`

Sample Function Call

```
http://www.company.com/cgi-bin/arswww.cgi?_function=printdocs  
&_server=od400&_user=web&_password=web  
&_folder=credit%20card%20statements  
&_printer=infoprint60  
&_docids=6850-6851-SUA17-1FAAA-225712-1634-132014-132172-89-76-11-25-0  
&_logoff=1
```


Funkce Retrieve Document (Načtení dokumentu)

Načte vybraný dokument ze serveru OnDemand.

Účel

Funkce Retrieve Document (Načtení dokumentu) načítá vybraný dokument ze serveru OnDemand. Aplikace ODWEK zobrazí dokument v okně prohlížeče pomocí appletu, prohlížeče nebo jiného programu, který je přiřazen k tomuto typu dokumentu.

Parametry

Tabulka 9. Funkce Retrieve Document

Jméno=hodnota	Účel
<code>_function=retrieve</code>	Načítá vybraný dokument.
<code>_server=hodnota</code>	Jméno serveru OnDemand.
<code>_user=hodnota</code>	ID uživatele OnDemand.
<code>_password=hodnota</code>	Heslo pro ID uživatele.
<code>_folder=hodnota</code>	Jméno složky.
<code>jméno pole složky=hodnota</code>	Jméno vyhledávacího pole složky a vyhledávací hodnota. Můžete zadat jednu nebo více sad jmen polí a vyhledávací hodnoty, a to až do počtu polí definovaných pro složku.
<code>_html=hodnota</code>	<p>Když dojde k chybě při načítání dokumentu, určuje soubor HTML, který aplikace ODWEK používá jako šablonu pro generování (chybové) výstupní webové stránky. Hodnota může být jméno souboru nebo * (hvězdička). Jestliže hodnota je hvězdička, pak aplikace ODWEK používá soubor RETRIEVE.HTML umístěný v adresáři zadaném v parametru <code>TEMPLATEDIR</code> v souboru <code>ARSWWW.INI</code>. Jestliže hodnota je jméno souboru bez jména cesty, soubor se musí nacházet v adresáři zadaném v parametru <code>TEMPLATEDIR</code>. Jestliže hodnota zahrnuje jméno cesty, pak by se cesta měla vztahovat k adresáři zadanému v parametru <code>TEMPLATEDIR</code>.</p> <p>Souhrnný obsah souboru HTML definuje zákazník. Soubor však musí obsahovat následující řádek komentáře:</p> <pre><!-- -AOI# Marker-- -></pre> <p>Umístění řádku komentáře určuje, kam aplikace ODWEK umísťuje svůj výstup. Všechny řádky uvedené nad řádkem komentáře se zapíší před výstup generovaný aplikací ODWEK. Všechny řádky uvedené pod řádkem komentáře se zapíší za výstup generovaný aplikací ODWEK.</p> <p>Soubor <code>TEMPLATE.HTM</code> je vzorový soubor šablony dodávaný aplikací ODWEK. Pomocí vzorového souboru šablony můžete vytvořit vlastní soubor šablony pro funkci načtení.</p>
<code>_nohtml=hodnota</code>	Určuje typ výstupu generovaného aplikací ODWEK. Předvolená hodnota je 0 (nula) a znamená, že aplikace ODWEK generuje výstup HTML. Jestliže zadáte hodnotu 1 (jedna), aplikace ODWEK generuje výstup ASCII s oddělovači. Podrobnosti o výstupu ASCII s oddělovači najdete v tématu Dodatek H, "Žádný HTML výstup", na stránce 131.
<code>_port=hodnota</code>	Číslo portu pro server OnDemand. Předvolená hodnota 0 (nula) znamená, že server používá číslo portu, které je uvedeno v tabulce služeb (<code>WRKSRVTBLE</code>). Jestliže v tabulce služeb není uvedeno číslo portu, server OnDemand se pokusí použít číslo portu 1445. Každá vámi zadaná hodnota potlačí hodnotu parametru <code>PORT</code> v souboru <code>ARSWWW.INI</code> .

Tabulka 9. Funkce Retrieve Document (pokračování)

Jméno=hodnota	Účel
_codepage=hodnota	Kódová stránka databáze OnDemand. Předvolená kódová stránka je kódová stránka HTTP serveru. Pokud kódová stránka serveru je jiná než kódová stránka databáze, musíte kódovou stránku zadat. Každá vámi zadaná hodnota potlačí hodnotu parametru CODEPAGE v souboru ARSWWW.INI.
_cgibin=program	Používá se programem CGI, když se generuje další výstupní strana. Pokud je zadáno, stránka bude obsahovat volání zadaného programu namísto předvoleného programu (ARSWWW.CGI). Tento parametr používají především programátoři, kteří vytvářejí přední program CGI nebo servlet pro program CGI nebo servlet dodávaný IBM. Parametr <i>program</i> může uvádět adresář, který se vztahuje k direktivě ServerRoot nebo jménu <i>alias</i> , které je definováno v konfiguračním souboru HTTP serveru. Aplikace ODWEK standardně načítá program CGI z adresáře CGI-BIN.
_or=hodnota	Chcete-li vyhledávací pole spojit logickým operátorem OR, zadejte 1 (jedna); položka musí odpovídat alespoň jedné ze zadaných vyhledávacích hodnot. Předvolená hodnota je 0 (nula), což znamená, že server OnDemand spojuje vyhledávací pole logickým operátorem AND (položka musí odpovídat všem zadaným vyhledávacím hodnotám).
_afp=hodnota	Když načítáte dokument AFP ze serveru OnDemand, hodnota tohoto parametru určuje, jakou akci - pokud vůbec nějakou - aplikace ODWEK provede, než dokument odešle klientovi. Někteří zákazníci například konvertují dokumenty AFP na dokumenty HTML pomocí služby AFP2WEB Transform a používají applet AFP2HTML k zobrazení HTML výstupu. Tito zákazníci by měli uvést _afp=HTML , aby aplikace ODWEK konvertovala dokument AFP před tím, než jej odešle klientovi. <i>Hodnota může být:</i> ASCII Aplikace ODWEK konvertuje dokument AFP na text ASCII. HTML Aplikace ODWEK konvertuje dokument AFP na HTML pomocí služby AFP2WEB Transform. NATIVE Aplikace ODWEK rozbaluje a dekomprimuje dokument AFP a jeho prostředky ze serveru OnDemand. Poznámka: Jestliže zadáte _afp=NATIVE , ověřte, aby typ obsahu MIME uváděl prohlížeč, který chcete používat. Další informace najdete v tématu "[MIMETYPES]" na stránce 22. PDF Aplikace ODWEK konvertuje dokument AFP na PDF pomocí služby AFP2WEB Transform. PLUGIN Aplikace ODWEK nekonvertuje dokument AFP (předvolba).
_email=hodnota	Když načítáte dokument EMAIL ze serveru OnDemand, hodnota tohoto parametru určuje, jakou akci - pokud vůbec nějakou - aplikace ODWEK provede, než dokument odešle klientovi. <i>Hodnota může být:</i> NATIVE Aplikace ODWEK rozbaluje a dekomprimuje dokument EMAIL ze serveru OnDemand. Poznámka: Jestliže zadáte _email=NATIVE , ověřte, aby typ obsahu MIME uváděl prohlížeč, který chcete používat (další informace najdete v tématu "[MIMETYPES]" na stránce 22). HTML Aplikace ODWEK konvertuje dokument EMAIL na HTML.

Tabulka 9. Funkce Retrieve Document (pokračování)

Jméno=hodnota	Účel
_line=hodnota	<p>Když načítáte dokument Line Data ze serveru OnDemand, hodnota tohoto parametru určuje, jakou akci - pokud vůbec nějakou - aplikace ODWEK provede, než dokument odešle klientovi. <i>Hodnota</i> může být:</p> <p>APPLET Aplikace ODWEK konvertuje dokument Line Data pro zobrazení pomocí appletu Line Data (předvolba).</p> <p>ASCII Aplikace ODWEK konvertuje dokument Line Data na ASCII text.</p> <p>NATIVE Aplikace ODWEK rozbaluje a dekomprimuje dokument Line Data ze serveru OnDemand.</p> <p>Poznámka: Jestliže zadáte _line=NATIVE, ověřte, aby typ obsahu MIME uváděl prohlížeč, který chcete používat (další informace najdete v tématu “[MIMETYPES]” na stránce 22).</p>
_docid=documentID	Identifikátor dokumentu, který se má načíst. Identifikátor dokumentů vrácený funkcí Document Hit List.
_logoff=1	Po načtení dokumentu automaticky odpojí uživatele od serveru OnDemand. Když zadáte tento parametr, není zapotřebí, aby vaše aplikace volala funkci Logoff pro odpojení uživatele. Jediná platná hodnota pro tento parametr je 1 (jedna).

Užití

Tyto parametry jsou povinné:

_function
 _server
 _user
 _password
 _folder

Tyto parametry jsou volitelné:

jméno pole složky
 _docid
 _or
 _afp
 _email
 _line
 _html
 _nohtml
 _port
 _codepage
 _logoff
 _cgibin

Sample Function Call

```

http://www.company.com/cgi-bin/arswww.cgi?_function=retrieve
&_server=od400&_user=web&_password=web
&_folder=credit%20card%20statements
&account%20number=1000100010009999&date=1%2f1%2f96
&_html=template.htm&_logoff=1
  
```

Funkce Search Criteria (Vyhledávací kritéria)

Zobrazení vyhledávacích kritérií pro určitou složku.

Účel

Funkce Search Criteria (Vyhledávací kritéria) zobrazuje vyhledávací kritéria pro určitou složku pomocí formuláře. Uživatel může přijmout předvolená vyhledávací kritéria nebo zadat vyhledávací kritéria pro vyhledání určitého dokumentu. Po kliknutí na tlačítko Submit (Odeslat) aplikace ODWEK zobrazí webovou stránku, která vypisuje dokumenty, které odpovídají vyhledávacím kritériím.

Parametry

Tabulka 10. Funkce Search Criteria

Jméno=hodnota	Účel
<code>_function=searchcrit</code>	Zobrazuje vyhledávací kritéria pro určitou složku.
<code>_server=hodnota</code>	Jméno serveru OnDemand.
<code>_user=hodnota</code>	ID uživatele OnDemand.
<code>_password=hodnota</code>	Heslo pro ID uživatele.
<code>_folder=hodnota</code>	Jméno prohledávané složky.
<code>_html=hodnota</code>	<p>Určuje soubor HTML, který aplikace ODWEK používá jako šablonu pro generování výstupní webové stránky. Hodnota může být jméno souboru nebo * (hvězdička). Jestliže hodnota je hvězdička, pak aplikace ODWEK používá soubor SEARCHCRIT.HTML umístěný v adresáři zadaném v parametru <code>TEMPLATEDIR</code> v souboru ARSWWW.INI. Jestliže hodnota je jméno souboru beze jména cesty, soubor se musí nacházet v adresáři zadaném v parametru <code>TEMPLATEDIR</code>. Jestliže hodnota zahrnuje jméno cesty, pak by se cesta měla vztahovat k adresáři zadanému v proměnné <code>TEMPLATEDIR</code>.</p> <p>Souhrnný obsah souboru HTML definuje zákazník. Soubor však musí obsahovat následující řádek komentáře:</p> <pre><!-- -AOI# Marker-- -></pre> <p>Umístění řádku komentáře určuje, kam aplikace ODWEK umísťuje svůj výstup. Všechny řádky uvedené nad řádkem komentáře se zapíší před výstup generovaný aplikací ODWEK. Všechny řádky uvedené pod řádkem komentáře se zapíší za výstup generovaný aplikací ODWEK.</p> <p>Soubor <code>TEMPLATE.HTM</code> je vzorový soubor šablony dodávaný aplikací ODWEK. Pomocí vzorového souboru šablony můžete vytvořit vlastní soubor šablony pro funkci vyhledávací kritéria.</p>
<code>_frame=hodnota</code>	Výstup tohoto příkazu bude zahrnovat atribut <code>target=hodnota</code> . Tento parametr usnadňuje vytváření rámců HTML. Jedná se o volitelný parametr.
<code>_datefmt=hodnota</code>	Určuje formát hodnoty data používaný aplikací ODWEK k zobrazení položek, které odpovídají kritériím dotazu. Předvolený formát data je nastaven na straně Field Information příslušné složky. Podrobnosti o formátech podporovaných na serveru OnDemand najdete v publikaci <i>IBM Content Manager OnDemand for iSeries V5R1 Common Server Administration Guide</i> , SC27-1161.
<code>_nohtml=hodnota</code>	Určuje typ výstupu generovaného aplikací ODWEK. Předvolená hodnota je 0 (nula) a znamená, že aplikace ODWEK generuje výstup HTML. Jestliže zadáte hodnotu 1 (jedna), aplikace ODWEK generuje výstup ASCII s oddělovači. Podrobnosti o výstupu ASCII s oddělovači najdete v tématu Dodatek H, "Žádný HTML výstup", na stránce 131.

Tabulka 10. Funkce Search Criteria (pokračování)

Jméno=hodnota	Účel
<code>_port=hodnota</code>	Číslo portu pro server OnDemand. Předvolená hodnota 0 (nula) znamená, že server používá číslo portu, které je uvedeno v tabulce služeb (WRKSRVTBLE). Jestliže v tabulce služeb není uvedeno číslo portu, server OnDemand se pokusí použít číslo portu 1445. Každá vámi zadaná hodnota potlačí hodnotu parametru PORT v souboru ARSWWW.INI.
<code>_codepage=hodnota</code>	Kódová stránka databáze OnDemand. Předvolená kódová stránka je kódová stránka HTTP serveru. Pokud kódová stránka serveru je jiná než kódová stránka databáze, musíte kódovou stránku zadat. Každá vámi zadaná hodnota potlačí hodnotu parametru CODEPAGE v souboru ARSWWW.INI.
<code>_cgibin=program</code>	Používá se programem CGI, když se generuje další výstupní strana. Pokud je zadáno, stránka bude obsahovat volání zadaného programu namísto předvoleného programu (ARSWWW.CGI). Tento parametr používají především programátoři, kteří vytvářejí přední program CGI nebo servlet pro program CGI nebo servlet dodávaný IBM. Parametr <i>program</i> může uvádět adresář, který se vztahuje k direktivě ServerRoot nebo jménu <i>alias</i> , které je definováno v konfiguračním souboru HTTP serveru. Aplikace ODWEK standardně načítá program CGI z adresáře CGI-BIN.
<code>_logoff=1</code>	Po zobrazení vyhledávacích kritérií automaticky odpojí uživatele od serveru OnDemand. Když zadáte tento parametr, není zapotřebí, aby vaše aplikace volala funkci Logoff pro odpojení uživatele. Jediná platná hodnota pro tento parametr je 1 (jedna).

Užití

Tyto parametry jsou povinné:

- `_function`
- `_server`
- `_user`
- `_password`
- `_folder`

Tyto parametry jsou volitelné:

- `_frame`
- `_datefmt`
- `_html`
- `_nohtml`
- `_port`
- `_codepage`
- `_logoff`
- `_cgibin`

Sample Function Call

```
http://www.company.com/cgi-bin/arswww.cgi?_function=searchcrit
&_server=od400&_user=web&_password=web
&_folder=credit%20card%20statements&_html=template.htm
&_logoff=1
```

Funkce Update Document (Aktualizace dokumentu)

Aktualizuje jednu nebo více databázových hodnot pro zadaný dokument.

Účel

Funkce Update Document (Aktualizace dokumentu) umožňuje oprávněným uživatelům aktualizovat dokumenty. Funkce Update Document (Aktualizace dokumentu) aktualizuje jednu nebo více databázových hodnot pro určitý dokument.

Parametry

Tabulka 11. Funkce Update Document

Jméno=hodnota	Účel
<code>_function=updatedoc</code>	Aktualizuje databázi.
<code>_server=hodnota</code>	Jméno serveru OnDemand.
<code>_user=hodnota</code>	ID uživatele OnDemand. Uživatel musí mít oprávnění Update document pro příslušnou skupinu aplikací.
<code>_password=hodnota</code>	Heslo pro uživatele.
<code>_folder=hodnota</code>	Jméno složky.
<code>jméno pole složky=hodnota</code>	Jméno pole, které chcete aktualizovat, a hodnota, kterou chcete zadat do pole. Můžete zadat jednu nebo více sad jmen polí a hodnoty, a to až do počtu polí definovaných pro složku.
<code>_html=hodnota</code>	<p>Určuje soubor HTML, který aplikace ODWEK používá jako šablonu pro generování výstupní webové stránky. Hodnota může být jméno souboru nebo * (hvězdička). Jestliže hodnota je hvězdička, pak aplikace ODWEK používá soubor UPDATE.HTML umístěný v adresáři zadaném v parametru <code>TEMPLATEDIR</code> v souboru ARSWWW.INI. Jestliže hodnota je jméno souboru bez jména cesty, soubor se musí nacházet v adresáři zadaném v parametru <code>TEMPLATEDIR</code>. Jestliže hodnota zahrnuje jméno cesty, pak by se cesta měla vztahovat k adresáři zadanému v parametru <code>TEMPLATEDIR</code>.</p> <p>Souhrnný obsah souboru HTML definuje zákazník. Soubor však musí obsahovat následující řádek komentáře:</p> <pre><!-- - -AOI# Marker-- - -></pre> <p>Umístění řádku komentáře určuje, kam aplikace ODWEK umísťuje svůj výstup. Všechny řádky uvedené nad řádkem komentáře se zapíší před výstup generovaný aplikací ODWEK. Všechny řádky uvedené pod řádkem komentáře se zapíší za výstup generovaný aplikací ODWEK.</p> <p>Soubor <code>TEMPLATE.HTM</code> je vzorový soubor šablony dodávaný aplikací ODWEK. Pomocí vzorového souboru šablony můžete vytvořit vlastní soubor šablony pro funkci aktualizace.</p>
<code>_nohtml=hodnota</code>	Určuje typ výstupu generovaného aplikací ODWEK. Předvolená hodnota je 0 (nula) a znamená, že aplikace ODWEK generuje výstup HTML. Jestliže zadáte hodnotu 1 (jedna), aplikace ODWEK generuje výstup ASCII s oddělovači. Podrobnosti o výstupu ASCII s oddělovači najdete v tématu Dodatek H, "Žádný HTML výstup", na stránce 131.
<code>_docid=documentID</code>	Identifikátor dokumentu, který se má aktualizovat. Identifikátor dokumentů vrácený funkcí Document Hit List.

Tabulka 11. Funkce Update Document (pokračování)

Jméno=hodnota	Účel
<code>_port=hodnota</code>	Číslo portu pro server OnDemand. Předvolená hodnota 0 (nula) znamená, že server používá číslo portu, které je uvedeno v tabulce služeb (WRKSRVTBLE). Jestliže v tabulce služeb není uvedeno číslo portu, server OnDemand se pokusí použít číslo portu 1445. Každá vámi zadaná hodnota potlačí hodnotu parametru PORT v souboru ARSWWW.INI.
<code>_codepage=hodnota</code>	Kódová stránka databáze OnDemand. Předvolená kódová stránka je kódová stránka HTTP serveru. Pokud kódová stránka serveru je jiná než kódová stránka databáze, musíte kódovou stránku zadat. Každá vámi zadaná hodnota potlačí hodnotu parametru CODEPAGE v souboru ARSWWW.INI.
<code>_logoff=1</code>	Po aktualizaci dokumentu automaticky odpojí uživatele od serveru OnDemand. Když zadáte tento parametr, není zapotřebí, aby vaše aplikace volala funkci Logoff pro odpojení uživatele. Jediná platná hodnota pro tento parametr je 1 (jedna).

Užití

Tyto parametry jsou povinné:

`_function`
`_server`
`_user`
`_password`
`_folder`

Tyto parametry jsou volitelné:

jméno pole složky
`_docid`
`_html`
`_nohtml`
`_port`
`_codepage`
`_logoff`

Sample Function Call

```
http://www.company.com/cgi-bin/arswww.cgi?_function=updatedoc
&_server=od400&_user=web&_password=web
&_folder=credit%20card%20statements
&account%20number=1000100010009999
&_docid=6850-6851-SUA17-1FAAA-225712-1634-132014-132172-89-76-11-25-0
&_html=template.htm&_logoff=1
```

Funkce Zobrazení anotací (View Annotations)

Zobrazení anotací připojených k zadanému dokumentu.

Účel

Funkce View Annotations (Zobrazení anotací) umožňuje uživatelům zobrazit anotace připojené k zadanému dokumentu. K zobrazení anotací musí mít uživatel oprávnění View Annotation ve skupině aplikací OnDemand. (Oprávnění Access také umožňuje uživatelům zobrazit anotace.)

Parametry

Tabulka 12. Funkce View Annotations

Jméno=hodnota	Účel
<code>_function=getnotes</code>	Zobrazuje anotace.
<code>_server=hodnota</code>	Jméno serveru OnDemand.
<code>_user=hodnota</code>	ID uživatele OnDemand. Uživatel musí mít oprávnění Annotation View pro každou skupinu aplikací, která obsahuje anotace, jež se mají zobrazit. (Oprávnění Application Group Access umožňuje uživatelům prohlížet anotace.)
<code>_password=hodnota</code>	Heslo pro uživatele.
<code>_folder=hodnota</code>	Jméno složky.
<code>_html=hodnota</code>	<p>Určuje soubor HTML, který aplikace ODWEK používá jako šablonu pro generování výstupní webové stránky. Hodnota může být jméno souboru nebo * (hvězdička). Jestliže hodnota je hvězdička, pak aplikace ODWEK používá soubor GETNOTES.HTML umístěný v adresáři zadaném v parametru <code>TEMPLATEDIR</code> v souboru ARSWWW.INI. Jestliže hodnota je jméno souboru beze jména cesty, soubor se musí nacházet v adresáři zadaném v parametru <code>TEMPLATEDIR</code>. Jestliže hodnota zahrnuje jméno cesty, pak by se cesta měla vztahovat k adresáři zadanému v parametru <code>TEMPLATEDIR</code>.</p> <p>Souhrnný obsah souboru HTML definuje zákazník. Soubor však musí obsahovat následující řádek komentáře:</p> <pre><!-- -AOI# Marker-- -></pre> <p>Umístění řádku komentáře určuje, kam aplikace ODWEK umísťuje svůj výstup. Všechny řádky uvedené nad řádkem komentáře se zapíší před výstup generovaný aplikací ODWEK. Všechny řádky uvedené pod řádkem komentáře se zapíší za výstup generovaný aplikací ODWEK.</p> <p>TEMPLATE.HTM je vzorový soubor šablony dodávaný aplikací ODWEK. Pomocí vzorového souboru šablony můžete vytvořit vlastní soubor šablony pro funkci zobrazení anotací.</p>
<code>_nohtml=hodnota</code>	Určuje typ výstupu generovaného aplikací ODWEK. Předvolená hodnota je 0 (nula) a znamená, že aplikace ODWEK generuje výstup HTML. Jestliže zadáte hodnotu 1 (jedna), aplikace ODWEK generuje výstup ASCII s oddělovači. Podrobnosti o výstupu ASCII s oddělovači najdete v tématu Dodatek H, "Žádný HTML výstup", na stránce 131.
<code>_docid=documentID</code>	Identifikátor dokumentu, který obsahuje anotace, jež se mají zobrazit. Identifikátor dokumentů vrácený funkcí Document Hit List.
<code>_port=hodnota</code>	Číslo portu pro server OnDemand. Předvolená hodnota 0 (nula) znamená, že server používá číslo portu, které je uvedeno v tabulce služeb (WRKSRVTBLE). Jestliže v tabulce služeb není uvedeno číslo portu, server OnDemand se pokusí použít číslo portu 1445. Každá vámi zadaná hodnota potlačí hodnotu parametru <code>PORT</code> v souboru ARSWWW.INI.

Tabulka 12. Funkce View Annotations (pokračování)

Jméno=hodnota	Účel
<code>_codepage=hodnota</code>	Kódová stránka databáze OnDemand. Předvolená kódová stránka je kódová stránka HTTP serveru. Pokud kódová stránka serveru je jiná než kódová stránka databáze, musíte kódovou stránku zadat. Každá vámi zadaná hodnota potlačí hodnotu parametru CODEPAGE v souboru ARSWWW.INI.
<code>_logoff=1</code>	Automaticky odpojuje uživatele od serveru OnDemand po zobrazení anotace. Když zadáte tento parametr, není zapotřebí, aby vaše aplikace volala funkci Logoff pro odpojení uživatele. Jediná platná hodnota pro tento parametr je 1 (jedna).

Užití

Tyto parametry jsou povinné:

- `_function`
- `_server`
- `_user`
- `_password`
- `_folder`
- `_docid`

Tyto parametry jsou volitelné:

- `_html`
- `_nohtml`
- `_port`
- `_codepage`
- `_logoff`

Sample Function Call

```
http://www.company.com/cgi-bin/arswww.cgi?_function=getnotes
&_server=od400&_user=web&_password=web
&_folder=credit%20card%20statements
&_docid=6850-6851-SUA17-1FAAA-225712-1634-132014-132172-89-76-11-25-0
&_logoff=1
```

Dodatek B. Reference pro Java servlet

Java servlet pracuje jako řadič vaší webové aplikace - provádí funkce a běžné úlohy před akcí a po akci, například spravuje připojení k serveru OnDemand.

Pro typické aplikační úlohy jsou k dispozici tyto funkce:

- přihlášení a odhlášení
- vyhledání
- načtení, tisk a aktualizace dokumentů
- přidání a prohlížení anotací
- změna hesla

Pomocí sady aplikačních funkcí a parametrů můžete používat servlet ve své aplikaci.

Java servlet používá stejné funkce jako program CGI. Informace o funkcích, popisech a parametrech najdete v tématu Dodatek A, "Referenční informace pro rozhraní CGI API", na stránce 53.

Dodatek C. Reference pro rozhraní Java API

Dokumentace pro rozhraní Java API se dodává ve formátu HTML se softwarem ODWEK.

Než budete moci zobrazit dokumentaci, musíte nainstalovat software ODWEK software na systém a pak rozbalit soubory dokumentace ze souboru ODApiDoc.zip v adresáři /QIBM/ProdData/OnDemand/www/. Použijte metodu extrakce, která uchová strukturu adresářů souborů v archivu.

Chcete-li zobrazit dokumentaci (pokud jste již rozbalili soubory), otevřete soubor index.html ve webovém prohlížeči.

Dodatek D. Průvodce programováním rozhraní Java API

Rozhraní Java API jsou sady tříd, které přistupují k datům a manipulují s nimi na serveru OnDemand. Tato část popisuje rozhraní Java API, implementaci dokumentových funkcí Java a internetovou konektivitu.

Rozhraní Java API podporují:

- Společný objektový model pro přístup k datům.
- Vyhledání a aktualizace na serverech OnDemand. **Poznámka:** Omezení při přístupu k serveru OnDemand pro server OS/390 verze 2 uvádí Kapitola 1, "Přehled", na stránce 1.
- Implementace klient/server pro uživatele aplikace Java.

Architektura klient/server

Rozhraní API poskytují pohodlné programovací rozhraní pro uživatele aplikací. Rozhraní API mohou být umístěna na serveru a klientovi OnDemand (obě možnosti poskytují stejné rozhraní), a aplikace mohou být umístěny lokálně nebo jinde (vzdálený přístup). Klient API komunikuje se serverem při přístupu k datům po síti. Komunikace mezi klientem a serverem se provádí podle tříd; není nutné přidávat žádné další programy.

Třídy API se skládají z jednoho balíku: `com.ibm.edms.od`.

Balení pro prostředí Java

Třídy rozhraní API jsou obsaženy v jednom balení: `com.ibm.edms.od`. Jsou to tyto třídy:

`com.ibm.edms.od.ODCallback`

Tato třída se používá se všemi metodami, během jejichž zpracování server vrací data.

`com.ibm.edms.od.ODCriteria`

Třída, která představuje kritéria vyhledávání ze složky OnDemand. Třída kritérií obsahuje metody pro nastavení operátora vyhledávání a hodnoty vyhledávání.

`com.ibm.edms.od.ODException`

Tato třída představuje výjimky, k nimž může dojít při používání rozhraní API.

`com.ibm.edms.od.ODFolder`

Třída, která představuje složku OnDemand. Tento objekt se vrací z úspěšného volání do `ODServer.openFolder()`. Tato třída obsahuje informace o kritériích složky. Tyto objekty kritérií jsou to, co je nutné pozměnit k zúžení dotazu na serveru.

`com.ibm.edms.od.ODHit`

Tato třída představuje dokument OnDemand.

`com.ibm.edms.od.ODNote`

Tato třída představuje anotaci OnDemand.

`com.ibm.edms.od.ODServer`

Tato třída představuje připojení k serveru OnDemand. Z této třídy se můžete přihlásit, odhlásit a měnit heslo. Po úspěšném přihlášení bude tento objekt obsahovat seznam všech složek, k nimž má relace přístup. **Poznámka:** Přístup k tomuto serverovému objektu by se měl provádět v jednovláknovém prostředí. Jediná výjimka je tehdy, když se ruší serverová operace.

Rady k programování

Balík `com.ibm.edms.od` musíte importovat do své aplikace ODWEK.

Není nutné, aby HTTP server nebo webový aplikační server spouštěl aplikace ODWEK, které používají rozhraní Java API. Překladač Java můžete spouštět na aplikacích ODWEK.

Spuštění překladače Java na aplikaci ODWEK:

1. Soubor `arswww.ini` zkopírujte do uživatelsky definovaného adresáře.
2. Sdílenou knihovnu zkopírujte do adresáře, do kterého jste nakopírovali soubor `arswww.ini`:

Tabulka 13. Jméno sdíleného souboru knihovny

Operační systém	Sdílená knihovna
AIX	<code>libarswwwsl.a</code>
HP-UX	<code>libarswwwsl.sl</code>
Linux	<code>libarswwwsl.so</code>
Solaris	<code>libarswwwsl.so</code>
Windows	<code>arswwwsl.dll</code>

3. V případě systémů Windows zkopírujte tyto soubory do adresáře, do kterého jste nakopírovali soubor `arswww.ini`:

`ARSSCKNT.DLL`
`ARSCT32.DLL`

4. Zadejte jméno uživatelsky definovaného adresáře, když na aplikaci spustíte překladač Java. Příklad najdete v tématu “Spuštění aplikace ODWEK” na stránce 89.

Nastavení systémové proměnné

Když nastavujete prostředí AIX, HP-UX, Linux, Solaris nebo Windows, musíte vytvořit následující nastavení:

package

Importujte všechny aplikace ODWEK.

- `com.ibm.edms.od`

Soubory knihoven

Sdílené objekty pro AIX, HP-UX, Linux a Solaris

Knihovny DLL pro Windows

Nastavení proměnných prostředí

Při vývoji aplikace ODWEK musíte nastavit své prostředí.

AIX

V prostředí AIX musíte nastavit následující proměnné svého vyvojářského prostředí pro tvorbu aplikací ODWEK.

PATH Zajistěte, aby vaše proměnná `PATH` obsahovala `/usr/lpp/ars/www`

LIBPATH Zajistěte, aby vaše proměnná `LIBPATH` obsahovala `/usr/lpp/ars/www`

LD_LIBRARY_PATH

Zajistěte, aby vaše proměnná `LD_LIBRARY_PATH` obsahovala `/usr/lpp/ars/www`

CLASSPATH Zajistěte, aby vaše proměnná **CLASSPATH** obsahovala /usr/lpp/ars/www/api/ODApi.jar, což je knihovna tříd.

HP-UX

V prostředí HP-UX musíte nastavit následující proměnné svého vyvojářského prostředí pro tvorbu aplikací ODWEK.

PATH Zajistěte, aby vaše proměnná **PATH** obsahovala /opt/ondemand/www

LIBPATH Zajistěte, aby vaše proměnná **LIBPATH** obsahovala /opt/ondemand/www

LD_LIBRARY_PATH

Zajistěte, aby vaše proměnná **LD_LIBRARY_PATH** obsahovala /opt/ondemand/www

CLASSPATH Zajistěte, aby vaše proměnná **CLASSPATH** obsahovala /opt/ondemand/www/api/ODApi.jar, což je knihovna tříd.

Linux

V prostředí Linux musíte nastavit následující proměnné svého vyvojářského prostředí pro tvorbu aplikací ODWEK.

PATH Zajistěte, aby vaše proměnná **PATH** obsahovala /opt/ondemand/www

LIBPATH Zajistěte, aby vaše proměnná **LIBPATH** obsahovala /opt/ondemand/www

LD_LIBRARY_PATH

Zajistěte, aby vaše proměnná **LD_LIBRARY_PATH** obsahovala /opt/ondemand/www

CLASSPATH Zajistěte, aby vaše proměnná **CLASSPATH** obsahovala /opt/ondemand/www/api/ODApi.jar, což je knihovna tříd.

Solaris

V prostředí Solaris musíte nastavit následující proměnné svého vyvojářského prostředí pro tvorbu aplikací ODWEK.

PATH Zajistěte, aby vaše proměnná **PATH** obsahovala /opt/ondemand/www

LIBPATH Zajistěte, aby vaše proměnná **LIBPATH** obsahovala /opt/ondemand/www

LD_LIBRARY_PATH

Zajistěte, aby vaše proměnná **LD_LIBRARY_PATH** obsahovala /opt/ondemand/www

CLASSPATH Zajistěte, aby vaše proměnná **CLASSPATH** obsahovala /opt/ondemand/www/api/ODApi.jar, což je knihovna tříd.

Windows

V prostředí Windows musíte nastavit následující proměnné svého vyvojářského prostředí pro tvorbu aplikací ODWEK.

PATH Zajistěte, aby **PATH** obsahovala x :\yyyyyyyy \DLL;, kde x je jednotka, na kterou jste instalovali ODWEK, a yyyyyyyy je instalační adresář pro software ODWEK.

CLASSPATH Zajistěte, aby proměnná **CLASSPATH** obsahovala x :\yyyyyyyy \WWW \API\ODApi.jar, kde x je jednotka, na kterou jste instalovali ODWEK, a yyyyyyyy je instalační adresář pro knihovnu tříd.

Trasování a diagnostické informace

K řešení problémů, k nimž dochází v rozhraní Java API vašich aplikací, můžete použít trasování a zacházení s výjimkami.

Trasování

Následující parametry v souboru ARSWWW.INI zapisují trasovací informace do souboru arswww.log v zadaném adresáři:

```
[DEBUG]
LOG=1
LOGDIR=/ars/www/log
```

Poznámka: Do souboru protokolu může být zapsáno významné množství informací, a proto IBM doporučuje, abyste protokolování povolovali pouze tehdy, když je to potřebné, například když zkoušíte znovu vyvolat problém. Pokud potřebujete povolit protokolování na delší dobu, ověřte, zda cesty protokolovacích souborů vedou na paměťová zařízení s dostatkem volného prostoru. Také pravidelně odstraňujte staré soubory protokolu ze systému.

Informace o jiných nástrojích, které můžete použít ke shromažďování informací o systému a dokumentech najdete v tématu Dodatek J, “Nástroje k určování problémů”, na stránce 137.

Zacházení s výjimkami

Když je v rozhraní Java API zaznamenán problém, vznikne výjimka. Výskyt výjimky vytvoří objekt výjimky třídy ODEException nebo některé z jejích podřízených tříd.

Když je vytvořen objekt ODEException, rozhraní API protokoluje diagnostické informace do souboru protokolu, přičemž se předpokládá, že protokolování je povoleno. Další informace o souboru protokolu použitého na rozhraní Java najdete v tématu “Trasování”.

Když je zachycen objekt ODEException, umožní vám to zobrazit libovolné chybové zprávy, chybové kódy a chybové stavy, k nimž došlo při zpracování. Když je zachycena chyba, bude vydána chybová zpráva uvádějící také místo, kde byla výjimka zachycena. Také je uveden ID chyby a ID výjimky. Níže uvedený kód je příklad procesu výskytu a zachycení.

```
try
{
    odServer = new ODServer( );
    odServer.initialize( argv[9], "TcUpdate.java" );
    System.out.println( "Přihlášení k " + argv[0] + "..." );
    odServer.logon( argv[0], argv[1], argv[2] );
    odServer.logoff( );
    odServer.terminate( );
}

catch ( ODEException e )
{
    System.out.println( "ODEException: " + e );
    System.out.println( "    id = " + e.getErrorId( ) );
    System.out.println( "    msg = " + e.getErrorMsg( ) );
    e.printStackTrace( );
}
```

Konstanty

Konstanty určené pro použití s rozhraními Java API jsou popsány v online referencích. Další informace najdete v tématu Dodatek C, “Reference pro rozhraní Java API”, na stránce 83.

Spuštění aplikace ODWEK

Překladač Java můžete použít ke spuštění aplikace ODWEK. Při vytváření, kompilaci a spouštění aplikace ODWEK pamatujte prosím na následující body:

1. Vytvořte aplikaci ODWEK pomocí metod, které máte k dispozici v rozhraní Java API. Importujte sadu rozhraní Java API do souboru své aplikace ODWEK. Příklad:

```
//*****  
import java.util.*;  
import java.io.*;  
import com.ibm.edms.od.*;  
  
public class Logon  
{  
    public static void main ( String argv[] )  
    {  
        .  
        .  
        .  
    }  
}
```

2. Kompilujte soubor své aplikace ODWEK (.java) s `javac` pro vytvoření souboru .class. Informace o kompilování aplikací Java najdete ve své referenční publikaci Java.
3. Spusťte překladač Java na svou aplikaci (soubor .class). Příklad:
`java Logon server userid passwd /tmp/ondemand/www`

Kde `Logon` je jméno souboru .class, `server`, `userid` a `passwd` jsou parametry pro aplikaci a `/tmp/ondemand/www` je uživatelsky definovaný runtime adresář, který obsahuje kopii souboru `arswww.ini`. **Poznámka:** Tento příklad předpokládá, že máte zadanou cestu ke třídě ODWEK a knihovnam servletů pomocí proměnných systémového prostředí (viz téma “Nastavení systémové proměnné” na stránce 86).

Připojování k serveru OnDemand

Objekt třídy `ODServer` představuje a spravuje připojení k serveru OnDemand, poskytuje podporu transakcí a spouští serverové příkazy. Dodatek C, “Reference pro rozhraní Java API”, na stránce 83 uvádí, kde lze nalézt online reference k metodám a jejich popisy.

Při připojování k serveru OnDemand musíte znát požadavky pro server; například heslo serveru OnDemand nesmí být delší než osm znaků.

Vytvoření připojení

Třída `ODServer` uvádí metody pro připojení k serveru OnDemand a pro odpojení od serveru. V následujícím příkladu je použit server knihoven OnDemand se jménem `LIBSRVR1`, ID uživatele `ADMIN` a heslem `PASSWD`. V tomto příkladu se vytvoří objekt `ODServer` pro server OnDemand, ustanoví se k němu připojení, pracuje se s ním (není v příkladu uvedeno) a pak se připojení ukončí.

```
odServer = new ODSERVER( );  
odServer.initialize( "c:\odwekdir", "Sample" );  
System.out.println( "Logging on to " + "LIBSRVR1" + "..." );  
odServer.logon( "LIBSRVR1", "ADMIN", "PASSWD" );  
.  
.  
.  
odServer.logoff( );  
odServer.terminate( );
```

Kompletní vzorovou aplikaci, z níž byl tento příklad převzat, najdete v tématu “Práce se serverem OnDemand” na stránce 90.

Nastavení a získání hesel

Metodami v objektu `ODServer` můžete přistupovat nebo nastavit hesla uživatele na objektu `OnDemand`. Následující příklad uvádí, jak lze nastavit a získat uživatelské heslo na serveru knihoven `OnDemand`.

```
odServer = new ODServer( );
odServer.setServer( "LIBSRVR1" );
odServer.setUserId( "ADMIN" );
odServer.setPassword( "PASSWD" );

System.out.println( "Logging on to " + "LIBSRVR1" + "..." );

odServer.logon( odServer.getServerName( ),
               odServer.getUserId( ),
               odServer.getPassword( ),
               ODConstant.CONNECT_TYPE_LOCAL,
               0 );
```

Kompletní vzorovou aplikaci, z níž byl tento příklad převzat, najdete v tématu “Práce se serverem `OnDemand`”.

Práce se serverem `OnDemand`

Objekt třídy `ODServer` představuje a spravuje připojení k serveru `OnDemand`, poskytuje podporu transakcí a spouští serverové příkazy.

V následujícím příkladu jsou použity metody `ODServer` pro přípravu k přihlášení, nastavení jména aplikace, (volitelné) zobrazení lokálního adresáře, zobrazení jména serveru, ID uživatele a hesla, zobrazení a nastavení typu připojení, zobrazení a nastavení portu a odpojení od serveru.

V tomto příkladu jsou předvedeny tyto metody `ODServer`:

- initialize
- logon
- logoff
- terminate
- getConnectType
- getLocalDir
- getPassword
- getPort
- getServerName
- getUserId
- setApplicationName
- setConnectType
- setLocalDir
- setPassword
- setPort
- setServer
- setUserId

V tomto příkladu jsou použity tyto runtime parametry:

- Jméno serveru
- ID uživatele
- Heslo
- Konfigurační adresář (umístění souboru `arswww.ini`)
- (volitelné) Adresář lokálního serveru

Příklad práce se serverem `OnDemand`:

```

//*****
import java.util.*;
import java.io.*;
import com.ibm.edms.od.*;

public class TcServerMisc
{
    public static void main ( String argv[] )
    {
        ODServer odServer;
        String str;
        int j;

        //-----
        // Je-li příliš málo parametrů, zobrazit syntaxi a odejít
        //-----
        if ( argv.length < 4 )
        {
            System.out.println( "usage: java TcServerMisc <server> <userid> <password> <config dir> [<local server dir>]" );
            return;
        }

        try
        {
            //-----
            // Vysvětlení
            //-----
            System.out.println( "Tento testovací případ by měl:" );
            System.out.println( " Použít metody ODServer setServer, setId a setPassword );
            System.out.println( " k přípravě pro přihlášení" );
            System.out.println( " Nastavení jména aplikace" );
            System.out.println( " Zobrazení" );
            System.out.println( " lokálního adresáře" );
            System.out.println( " jména serveru" );
            System.out.println( " ID uživatele" );
            System.out.println( " hesla" );
            System.out.println( " typu připojení" );
            System.out.println( " Nastavení a zobrazení portu" );
            System.out.println( " Nastavení typu připojení" );
            System.out.println( " Odhlášení" );
            System.out.println( "" );
            System.out.println( "Ověřte, zda všechny informace jsou správné." );
            System.out.println( "" );
            System.out.println( "-----" );
            System.out.println( "" );

            //-----
            // Přihlášení k zadanému serveru
            //-----
            odServer = new ODServer( );
            odServer.initialize( argv[3], "TcServerMisc.java" );
            odServer.setServer( argv[0] );
            odServer.setId( argv[1] );
            odServer.setPassword( argv[2] );

            System.out.println( "Přihlášení k " + argv[0] + "..." );
            if ( argv.length == 4 )
                odServer.logon( );
            else
            {
                if ( argv.length == 5 )
                {
                    odServer.setLocalDir( argv[4] );
                    odServer.logon( odServer.getServerName( ),
                                    odServer.getId( ),
                                    odServer.getPassword( ),
                                    ODConstant.CONNECT_TYPE_LOCAL,
                                    0,
                                    odServer.getLocalDir( ) );
                }
            }

            //-----
            // Testování různých metod
            //-----
            System.out.println( "Nastavení jména aplikace na TcServerMisc.java..." );
            odServer.setApplicationName( "TcServerMisc.java" );

            System.out.println( "Lokální adresář: " + odServer.getLocalDir( ) );
            System.out.println( "Jméno serveru: " + odServer.getServerName( ) );
            System.out.println( "ID uživatele: " + odServer.getId( ) );
            System.out.println( "Heslo: " + odServer.getPassword( ) );
            System.out.println( "Typ připojení: " + getConnectTypeName( odServer.getConnectType( ) ) );

            j = odServer.getPort( );
            System.out.println( "Nastavení portu na " + j + "..." );
            odServer.setPort( j );
            System.out.println( "Port: " + j );

            if ( argv.length == 4 )
            {
                System.out.println( "Nastavení typu připojení na ODConstant.CONNECT_TYPE_TCPIP..." );
                odServer.setConnectType( ODConstant.CONNECT_TYPE_TCPIP );
            }
        }
    }
}

```

```

else
{
    System.out.println( "Nastavení typu připojení na ODConstant.CONNECT_TYPE_LOCAL..." );
    odServer.setConnectType( ODConstant.CONNECT_TYPE_LOCAL );
}

//-----
// Čištění
//-----
System.out.println( "Odhlášení..." );
odServer.logoff( );
odServer.terminate( );
System.out.println( "" );
System.out.println( "-----" );
System.out.println( "" );
System.out.println( "Testovací případ dokončen - analyzujte, jestliže je to zapotřebí" );
System.out.println( "" );
}

catch ( ODEException e )
{
    System.out.println( "ODEException: " + e );
    System.out.println( "    id = " + e.getErrorId( ) );
    System.out.println( "    msg = " + e.getErrorMsg( ) );
    e.printStackTrace( );
}

catch ( Exception e2 )
{
    System.out.println( "výjimka: " + e2 );
    e2.printStackTrace( );
}
}

static String getConnectTypeName( char type )
{
    String str;

    switch( type )
    {
        case ODConstant.CONNECT_TYPE_TCPIP:
            str = "TCP/IP";
            break;
        case ODConstant.CONNECT_TYPE_LOCAL:
            str = "LOCAL";
            break;
        default:
            str = "*** Neznámý typ připojení";
            break;
    }

    return str;
}
}

```

Výpis skupin aplikací ve složce

Objekt třídy `ODFolder` představuje složku `OnDemand`.

V následujícím příkladu jsou použity metody `ODFolder` pro zobrazení počtu skupin aplikací, které lze najít ve složce, a pro zobrazení jména každé skupiny aplikací.

Tento příklad ukazuje použití těchto metod `ODFolder`:

- `getNumApplGroups`
- `getApplGroups`
- `close`

Tento příklad také používá metody `ODServer` pro přípravu k přihlášení, otevření zadané složky a odhlášení. V tomto příkladu jsou předvedeny tyto metody `ODServer`:

- `initialize`
- `logon`
- `openFolder`
- `logoff`
- `terminate`

V tomto příkladu jsou použity tyto runtime parametry:

- Jméno serveru
- ID uživatele

- Heslo
- Jméno složky
- Konfigurační adresář (umístění souboru arswwww.ini)
- (volitelné) Adresář lokálního serveru

Příklad výpisu skupina aplikací ve složce:

```

//*****
import java.util.*;
import java.io.*;
import com.ibm.edms.od.*;

public class TcAppGrp
{
    public static void main ( String argv[] )
    {
        ODServer odServer;
        ODFolder odFolder;
        Object[] appl_grps;
        int j;

        //-----
        // Je-li příliš málo parametrů, zobrazit syntaxi a odejít
        //-----
        if ( argv.length < 5 )
        {
            System.out.println( "použití: java TcAppGrp <server> <userid> <heslo> <folder> <config dir>
            [<lokální adresář serveru>]" );
            return;
        }

        try
        {
            //-----
            // Vysvětlení
            //-----
            System.out.println( "Tento testovací případ by měl:" );
            System.out.println( "  Přihlášení k zadanému serveru" );
            System.out.println( "  Otevření zadané složky" );
            System.out.println( "  Zobrazení jména složky" );
            System.out.println( "  Zobrazení počtu skupin aplikací" );
            System.out.println( "  Zobrazení jména každé skupina aplikací" );
            System.out.println( "  " );
            System.out.println( "-----" );
            System.out.println( "  " );

            //-----
            // Přihlášení k zadanému serveru
            //-----
            odServer = new ODServer( );
            odServer.initialize( argv[4], "TcListCriteria.java" );

            System.out.println( "Přihlášení k " + argv[0] + "..." );
            if ( argv.length == 5 )
                odServer.logon( argv[0], argv[1], argv[2] );
            else
                if ( argv.length == 6 )
                    odServer.logon( argv[0], argv[1], argv[2], ODConstant.CONNECT_TYPE_LOCAL, 0, argv[5] );

            //-----
            // Otevření zadané složky
            //-----
            System.out.println( "Otevírání " + argv[3] + " složka..." );
            odFolder = odServer.openFolder( argv[3] );

            //-----
            // Zobrazení počtu a jmen skupina aplikací
            //-----
            System.out.println( "Existuje " + odFolder.getNumAppGroups( ) + " skupin aplikací ve složce:" );
            appl_grps = odFolder.getAppGroups( );
            for ( j = 0; j < appl_grps.length; j++ )
                System.out.println( "  " + appl_grps[j].toString( ) );

            //-----
            // Čištění
            //-----
            odFolder.close( );
            odServer.logoff( );
            odServer.terminate( );
            System.out.println( "  " );
            System.out.println( "-----" );
            System.out.println( "  " );
            System.out.println( "Testovací případ dokončen - analyzujte výsledky, jestliže je to zapotřebí" );
            System.out.println( "  " );
        }

        catch ( ODEException e )
        {
            System.out.println( "ODEException: " + e );
            System.out.println( "  id = " + e.getErrorId( ) );
            System.out.println( "  msg = " + e.getErrorMsg( ) );
            e.printStackTrace( );
        }

        catch ( Exception e2 )

```

```

    {
        System.out.println( "výjimka: " + e2 );
        e2.printStackTrace( );
    }
}

```

Hledání složky

Objekt třídy `ODFolder` představuje složku `OnDemand`. Objekt třídy `ODCriteria` představuje hledací kritéria pro složku `OnDemand`. Objekt třídy `ODHit` představuje dokument `OnDemand`.

Následující příkaz používá metody `ODFolder` pro otevření zadané složky, zobrazení jména složky, popisu, pořadí zobrazení a vyhledávací kritéria, vyhledání složky a zavření složky. Tento příkaz používá metody `ODCriteria` k nastavení aktuálního vyhledávacího operandu a hledacích hodnot. Tento příklad používá metody `ODHit` ke zjištění zobrazovacích hodnot pro dokument, zjištění typu dokumentu, zjištění trvalého identifikátoru pro dokument, zjištění umístění dokumentu a zjištění typu obsahu MIME pro dokument.

Tento příklad ukazuje použití těchto metod `ODFolder`:

- `getName`
- `getDescription`
- `getDisplayOrder`
- `getCriteria`
- `search`
- `getSearchMessage`
- `close`

Tento příklad ukazuje použití těchto metod `ODCriteria`:

- `getName`
- `setOperand`
- `setSearchValue`
- `setSearchValues`

V tomto příkladu jsou předvedeny tyto metody `ODHit`:

- `getDisplayValue`
- `getDisplayValues`
- `getDocType`
- `getMimeType`
- `getDocLocation`
- `getDocId`

Tento příklad také používá metody `ODServer` pro přípravu k přihlášení, otevření zadané složky a odhlášení. V tomto příkladu jsou předvedeny tyto metody `ODServer`:

- `initialize`
- `logon`
- `openFolder`
- `terminate`

V tomto příkladu jsou použity tyto runtime parametry:

- Jméno serveru
- ID uživatele
- Heslo
- Jméno složky
- Jméno kritéria
- Operátor (musí být jeden z `eq`, `ne`, `lt`, `le`, `gt`, `ge`, `in`, `ni`, `li`, `nl`, `be`, `nb`)
- Hledací hodnota `l`

- (Volitelná) hledací hodnota 2
- Konfigurační adresář (umístění souboru arswwww.ini)

Poznámka: Počet nalezených výskytů může být omezen parametrem MAXHITS v souboru arswwww.ini.

Příklad hledání ve složce:

```
//*****
import java.util.*;
import java.io.*;
import com.ibm.edms.od.*;

public class TcSearch
{
    public static void main ( String argv[] )
    {
        ODServer odServer;
        ODFolder odFolder;
        ODCriteria odCrit;
        ODHit odHit;
        Enumeration values_enum;
        Vector hits;
        String[] display_crit;
        String header, line1, line2, hit_value, useable_value;
        boolean mismatch_detected;
        int j, k, opr;

        //-----
        // Je-li příliš málo parametrů, zobrazit syntaxi a odejít
        //-----
        if ( argv.length < 9 )
        {
            System.out.println( "použití: java TcSearch <server> <ID uživatele> <heslo> <složka> <kritéria> <opr> <hodnota1> <hodnota2>
            <konfigurační adresář>" );
            return;
        }

        try
        {
            //-----
            // Vysvětlení
            //-----
            System.out.println( "Tento testovací případ by měl:" );
            System.out.println( " Přihlášení k zadanému serveru" );
            System.out.println( " Otevření zadané složky" );
            System.out.println( " Zobrazení jména a popisu složky" );
            System.out.println( " Zjištění zadaného kritéria" );
            System.out.println( " Nastavení operátoru" );
            System.out.println( " Nastavení operandu (operandů)" );
            System.out.println( " Vyhledání složky" );
            System.out.println( " Zobrazení hledací zprávy (pokud se požaduje)" );
            System.out.println( " Zobrazení počtu výskytů" );
            System.out.println( " Zobrazení seznamu výskytů s každým výskytem na 3 řádcích:" );
            System.out.println( " 1. Hodnoty výskytů vrácené metodou ODHit.getDisplayValue" );
            System.out.println( " 2. Hodnoty výskytů vrácené metodou ODHit.getDisplayValues" );
            System.out.println( " 3. Typ dokumentu, typ MIME, umístění dokumentu, a hodnoty ID dokumentů" );
            System.out.println( "" );
            System.out.println( "Zajistěte, aby řádky 1 a 2 ze seznamu výskytů byly stejné a aby" );
            System.out.println( "hodnoty seznamu výskytů byly stejné jako ty, které se zobrazují v klientovi Windows." );
            System.out.println( "Jestliže soubor arswwww.ini omezuje počet výskytů, může být méně" );
            System.out.println( "výskytů, než se zobrazuje klientem Windows." );
            System.out.println( "" );
            System.out.println( "-----" );
            System.out.println( "" );

            //-----
            // Přihlášení k zadanému serveru
            //-----
            odServer = new ODServer( );
            odServer.initialize( argv[8], "TcSearch.java" );
            System.out.println( "Přihlášení k " + argv[0] + "..." );
            odServer.logon( argv[0], argv[1], argv[2] );

            //-----
            // Otevření zadané složky a vyhledání požadovaných kritérií
            //-----
            System.out.println( "Otevírání " + argv[3] + " složka..." );
            odFolder = odServer.openFolder( argv[3] );
            System.out.println( "Jméno=" + odFolder.getName( ) + " Popis=" + odFolder.getDescription( ) + " " );
            System.out.println( "Zjištění " + argv[4] + " kritéria..." );
            odCrit = odFolder.getCriteria( argv[4] );

            //-----
            // Konverze parametru operátoru na hodnotu interního operátoru a nastavení
            // operátoru kritérií
            //-----
            System.out.println( "Nastavení operátoru na " + argv[5] + "..." );
            if ( argv[5].equals( "eq" ) )
                opr = ODCONSTANT.OPEqual;
            else if ( argv[5].equals( "ne" ) )
                opr = ODCONSTANT.OPNotEqual;
            else if ( argv[5].equals( "lt" ) )
                opr = ODCONSTANT.OPLessThan;
            else if ( argv[5].equals( "le" ) )
                opr = ODCONSTANT.OPLessThanEqual;
            else if ( argv[5].equals( "gt" ) )
                opr = ODCONSTANT.OPGreaterThan;
            else if ( argv[5].equals( "ge" ) )
                opr = ODCONSTANT.OPGreaterThanEqual;
            else if ( argv[5].equals( "in" ) )
                opr = ODCONSTANT.OPIn;
            else if ( argv[5].equals( "ni" ) )
                opr = ODCONSTANT.OPNotIn;
        }
    }
}
```

```

        opr = ODConstant.OPNotIn;
    else if ( argv[5].equals( "li" ) )
        opr = ODConstant.OPLike;
    else if ( argv[5].equals( "nl" ) )
        opr = ODConstant.OPNotLike;
    else if ( argv[5].equals( "be" ) )
        opr = ODConstant.OPBetween;
    else if ( argv[5].equals( "nb" ) )
        opr = ODConstant.OPNotBetween;
    else
        opr = -1;

    System.out.println( " Nastavení operandu (operandů)..." );
    odCrit.setOperand( opr );

    if ( opr == ODConstant.OPBetween || opr == ODConstant.OPNotBetween )
    {
        odCrit.setSearchValues( argv[6], argv[7] );
        System.out.println( " " + odCrit.getName() + " " + getOperatorName( opr ) + " " + argv[6] + " a " + argv[7] );
    }
    else
    {
        odCrit.setSearchValue( argv[6] );
        System.out.println( " " + odCrit.getName() + " " + getOperatorName( opr ) + " " + argv[6] );
    }

    //-----
    // Prohledání složky
    //-----
    System.out.println( "Prohledávání " + argv[3] + "..." );
    hits = odFolder.search( );
    System.out.println( " Hledací zpráva: " + odFolder.getSearchMessage( ) );
    System.out.println( " Počet výskytů: " + hits.size( ) );

    //-----
    // Zobrazení výskytů
    //-----
    mismatch_detected = false;
    if ( hits != null && hits.size( ) > 0 )
    {
        display_crit = odFolder.getDisplayOrder( );
        header = " ";
        for( j = 0; j < display_crit.length; j++ )
            header = header + display_crit[j] + "--";
        System.out.println( "-----" );
        System.out.println( hlavička + " (z metody ODHit.getDisplayValue)" );
        System.out.println( hlavička + " (z metody ODHit.getDisplayValues)" );
        System.out.println( " DocType--MimeType--DocLocation--DocId" );
        System.out.println( "-----" );
        for ( j = 0; j < hits.size( ); j++ )
        {
            odHit = (ODHit)hits.elementAt( j );
            line1 = " ";
            for ( k = 0; k < display_crit.length; k++ )
            {
                hit_value = odHit.getDisplayValue( display_crit[k] );
                useable_value = ( hit_value.equals( "" ) ) ? " " : hit_value;
                line1 = line1 + useable_value + "--";
            }
            System.out.println( line1 );
            line2 = " ";
            for ( values_enum = odHit.getDisplayValues( ); values_enum.hasMoreElements( ); )
            {
                hit_value = (String)values_enum.nextElement( );
                useable_value = ( hit_value.equals( "" ) ) ? " " : hit_value;
                line2 = line2 + useable_value + "--";
            }
            System.out.println( line2 );
            System.out.println( " " + getDocTypeString( odHit.getDocType( ) ) +
                "--" + odHit.getMimeType( ) +
                "--" + getLocationString( odHit.getDocLocation( ) ) +
                "--" + odHit.getDocId( ) );
            if ( !line1.equals( line2 ) )
                mismatch_detected = true;
        }
    }

    //-----
    // Čištění
    //-----
    odFolder.close( );
    odServer.logoff( );
    odServer.terminate( );
    System.out.println( "" );
    System.out.println( "-----" );
    System.out.println( "" );
    System.out.println( "Testovací případ dokončen - analyzujte, jestliže je to zapotřebí" );
    System.out.println( "" );
    if ( mismatch_detected )
    {
        System.out.println( "*** Byla nalezena alespoň jedna neshoda mezi" );
        System.out.println( "*** řádky 1 a 2 ve výskytu" );
        System.out.println( "" );
    }
}

catch ( ODException e )
{
    System.out.println( "ODException: " + e );
    System.out.println( " id = " + e.getErrorId( ) );
    System.out.println( " msg = " + e.getErrorMsg( ) );
    e.printStackTrace( );
}

catch ( Exception e2 )
{
    System.out.println( "výjimka: " + e2 );
    e2.printStackTrace( );
}
}

```

```

static String getOperatorName( int oper )
{
    String str;

    switch( oper )
    {
        case ODConstant.OPEqual:
            str = "Equals";
            break;
        case ODConstant.OPNotEqual:
            str = "Not Equal";
            break;
        case ODConstant.OPLessThan:
            str = "Less Than";
            break;
        case ODConstant.OPLessThanEqual:
            str = "Less Than or Equal";
            break;
        case ODConstant.OPGreaterThan:
            str = "Greater Than";
            break;
        case ODConstant.OPGreaterThanEqual:
            str = "Greater Than or Equal";
            break;
        case ODConstant.OPIn:
            str = "In";
            break;
        case ODConstant.OPNotIn:
            str = "Not In";
            break;
        case ODConstant.OPLike:
            str = "Like";
            break;
        case ODConstant.OPNotLike:
            str = "Not Like";
            break;
        case ODConstant.OPBetween:
            str = "Between";
            break;
        case ODConstant.OPNotBetween:
            str = "Not Between";
            break;
        default:
            str = "Operator unknown";
            break;
    }

    return str;
}

static String getDocTypeString( char type )
{
    String str;

    switch( type )
    {
        case ODConstant.FileTypeAFP:
            str = "AFP";
            break;
        case ODConstant.FileTypeBMP:
            str = "BMP";
            break;
        case ODConstant.FileTypeEMAIL:
            str = "EMAIL";
            break;
        case ODConstant.FileTypeGIF:
            str = "GIF";
            break;
        case ODConstant.FileTypeJFIF:
            str = "JFIF";
            break;
        case ODConstant.FileTypeLINE:
            str = "LINE";
            break;
        case ODConstant.FileTypeMETA:
            str = "META";
            break;
        case ODConstant.FileTypeNONE:
            str = "NONE";
            break;
        case ODConstant.FileTypePCX:
            str = "PCX";
            break;
        case ODConstant.FileTypePDF:
            str = "PDF";
            break;
        case ODConstant.FileTypePNG:
            str = "PNG";
            break;
        case ODConstant.FileTypeTIFF:
            str = "TIFF";
            break;
        case ODConstant.FileTypeUSRDEF:
            str = "USRDEF";
            break;
        default:
            str = "*** Neplatný typ dokumentu ***";
            break;
    }

    return str;
}

static String getLocationString( int loc )
{
    String str;

    switch( loc )
    {

```

```

        case ODCConstant.DocLocationCache:
            str = "Cache";
            break;
        case ODCConstant.DocLocationArchive:
            str = "Archive";
            break;
        case ODCConstant.DocLocationExternal:
            str = "External";
            break;
        case ODCConstant.DocLocationUnknown:
            str = "Unknown";
            break;
        default:
            str = "*** Neplatné umístění dokumentu ***";
            break;
    }
    return str;
}

```

Vyhledání složky pomocí řetězce SQL

Následující příklad používá metody ODFolder k otevření zadané složky, prohledání složky se zadaným řetězcem SQL a zavřením složky. V tomto příkladu jsou použity metody ODHit k zobrazení počtu položek, které odpovídají dotazu, a k zobrazení seznamu dokumentů.

Tento příklad ukazuje použití těchto metod ODFolder:

- setApplGroupForSearchWithSQL
- search
- getDisplayOrder
- close

V tomto příkladu jsou předvedeny tyto metody ODHit:

- getDisplayValue

Tento příklad také používá metody ODServer pro přípravu k přihlášení, otevření zadané složky a odhlášení. V tomto příkladu jsou předvedeny tyto metody ODServer:

- initialize
- logon
- openFolder
- logoff
- terminate

V tomto příkladu jsou použity tyto runtime parametry:

- Jméno serveru
- ID uživatele
- Heslo
- Jméno složky
- Jméno skupiny aplikací
- řetězec SQL
- Konfigurační adresář (umístění souboru arswww.ini)

Příklad prohledání složky pomocí řetězce SQL:

```

//*****
//
// Testovací případ: TcSearchWithSQL
//
// Tento testovací případ provádí toto:
// Přihlášení k zadanému serveru
// Otevření zadané složky
// Prohledání složky s řetězcem SQL
// Zobrazení počtu výskytů
// Zobrazení seznamu výskytů
//
// Testuje následující metody:
// ODServer
// initialize
// logon
// openFolder
// logoff
// terminate
// ODFolder
// setApplGroupForSearchWithSQL
// search
// getDisplayOrder

```

```

// close
// ODHit
// getDisplayValue
//
// Parametry:
// 1. Jméno serveru
// 2. ID uživatele
// 3. Heslo
// 4. Jméno složky
// 5. Jméno skupiny aplikací
// 6. Řetězec SQL
// 7. Konfigurační adresář (obsahuje arswww.ini)
//
//*****
import java.util.*;
import java.io.*;
import com.ibm.edms.od.*;

public class TcSearchWithSQL
{
    public static void main ( String argv[] )
    {
        ODServer odServer;
        ODFolder odFolder;
        ODHit odHit;
        Enumeration values_enum;
        Vector hits;
        String[] display_crit;
        String server, userid, password, folder, directory;
        String sql, appl_group;
        String header, line, hit_value, useable_value;
        int j, k;

        //-----
        // Je-li příliš málo parametrů, zobrazit syntaxi a odejít
        //-----
        if ( argv.length < 6 )
        {
            System.out.println( "použití: java TcSearchWithSQL <server> <ID uživatele> <heslo> <složka> <skupina aplikací> <řetězec sql>
            <konfigurační adresář>" );
            return;
        }

        try
        {
            //-----
            // Vysvětlení
            //-----
            System.out.println( "Tento testovací případ by měl:" );
            System.out.println( " Přihlášení k zadanému serveru" );
            System.out.println( " Otevření zadané složky" );
            System.out.println( " Vyhledání složky se zadaným řetězcem SQL" );
            System.out.println( " Zobrazení počtu výskytů" );
            System.out.println( " Zobrazení seznamu výskytů" );
            System.out.println( "" );
            System.out.println( "-----" );
            System.out.println( "" );

            //-----
            // Přihlášení k zadanému serveru
            //-----
            server = argv[0];
            userid = argv[1];
            password = argv[2];
            folder = argv[3];
            appl_group = argv[4];
            sql = argv[5];
            directory = argv[6];

            odServer = new ODServer( );
            odServer.initialize( directory, "TcSearchWithSQL.java" );
            System.out.println( "Přihlášení k " + server + "..." );
            odServer.logon( server, userid, password );

            //-----
            // Otevření zadané složky
            //-----
            System.out.println( "Otevírání " + složka + " složka..." );
            odFolder = odServer.openFolder( folder );

            //-----
            // Prohledání složky
            //-----
            if ( appl_group.length( ) > 0 )
            {
                System.out.println( "Nastavení hledané skupiny aplikací: " + appl_group );
                odFolder.setAppGroupForSearchWithSQL( appl_group );
            }

            //-----
            // Prohledání složky
            //-----
            System.out.println( " Hledání " + složka + "..." );
            hits = odFolder.search( sql );
            System.out.println( " Počet výskytů: " + hits.size( ) );

            //-----
            // Zobrazení výskytů
            //-----
            if ( hits != null && hits.size( ) > 0 )
            {
                display_crit = odFolder.getDisplayOrder( );
                header = " ";
                for( j = 0; j < display_crit.length; j++ )
                    header = header + display_crit[j] + "--";
                System.out.println( "-----" );
                System.out.println( hlavička );
                System.out.println( "-----" );
                for ( j = 0; j < hits.size( ); j++ )

```

```

        {
            odHit = (ODHit)hits.elementAt( j );
            line = " ";
            for ( k = 0; k < display_crit.length; k++ )
            {
                hit_value = odHit.getDisplayValue( display_crit[k] );
                useable_value = ( hit_value.equals( "" ) ) ? " " : hit_value;
                line = line + useable_value + "--";
            }
            System.out.println( line );
        }
    }

    //-----
    // Čištění
    //-----
    odFolder.close( );
    odServer.logoff( );
    odServer.terminate( );
    System.out.println( "" );
    System.out.println( "-----" );
    System.out.println( "" );
    System.out.println( "Testovací případ dokončen - analyzujte, jestliže je to zapotřebí" );
    System.out.println( "" );
}

catch ( ODException e )
{
    System.out.println( "ODException: " + e );
    System.out.println( " id = " + e.getErrorId( ) );
    System.out.println( " msg = " + e.getErrorMsg( ) );
    e.printStackTrace( );
}

catch ( Exception e2 )
{
    System.out.println( "výjimka: " + e2 );
    e2.printStackTrace( );
}
}
}

```

Zrušení hledání

Následující příklad používá metodu `ODServer.cancel` ke zrušení probíhajícího hledání.

V tomto příkladu jsou použity metody `ODServer`, `ODFolder` a `ODCriteria` pro přihlášení ke serveru, otevření složky a nastavení kritéria `Date` na hodnotu 1970-2001. Druhé vlákno se pak inicializuje k provedení vyhledání. Když druhé vlákno skončí, zobrazí se počet nalezených výskytů. Pak se znovu inicializuje druhé vlákno k provedení vyhledání. Proces je uspán na 0.5 sekundy a vyhledání se pak zruší. Když druhé vlákno skončí, zobrazí se počet nalezených výskytů.

V tomto příkladu jsou předvedeny tyto metody `ODServer`:

- initialize
- logon
- openFolder
- logoff
- terminate

Tento příklad ukazuje použití těchto metod `ODFolder`:

- getCriteria
- search
- close

Tento příklad ukazuje použití těchto metod `ODCriteria`:

- setOperand
- setSearchValues

V tomto příkladu jsou použity tyto runtime parametry:

- Jméno serveru
- ID uživatele
- Heslo
- Jméno složky
- Konfigurační adresář (umístění souboru `arswww.ini`)
- (volitelné) Adresář lokálního serveru

Příklad zrušení vyhledání:

```
//*****
import java.util.*;
import java.io.*;
import com.ibm.edms.od.*;

class TestThread extends Thread
{
    ODFolder odFolder;

    TestThread( ODFolder fld )
    {
        odFolder = fld;
    }

    public void run( )
    {
        Vector hits;

        try
        {
            System.out.println( " Druhé vlákno hledá..." );
            hits = odFolder.search( );
            System.out.println( " Vyhledávání dokončeno - počet výskytů: " + hits.size( ) );
        }

        catch ( ODErrorException e )
        {
            System.out.println( "ODErrorException: " + e );
            System.out.println( " id = " + e.getErrorId( ) );
            System.out.println( " msg = " + e.getErrorMsg( ) );
            e.printStackTrace( );
        }

        catch ( Exception e2 )
        {
            System.out.println( "výjimka: " + e2 );
            e2.printStackTrace( );
        }
    }
}

public class TcCancelSearch
{
    public static void main ( String argv[] )
    {
        ODServer odServer;
        ODFolder odFolder;
        ODCriteria odCrit;
        TestThread search_thread;
        int j;

        //-----
        // Je-li příliš málo parametrů, zobrazit syntaxi a odejít
        //-----
        if ( argv.length < 5 )
        {
            System.out.println( "použití: java TcCancelSearch <server> <userid> <heslo> <folder> <config dir>
[<lokální adresář serveru>]" );
            return;
        }

        try
        {
            //-----
            // Vysvětlení
            //-----
            System.out.println( "Tento testovací případ by měl:" );
            System.out.println( " Přihlášení k zadanému serveru" );
            System.out.println( " Otevření zadané složky" );
            System.out.println( " Nastavení kritéria Date na hodnotu 1970-2001" );
            System.out.println( " Inicializace druhého vlákna k provedení vyhledání" );
            System.out.println( " Po dokončení druhého vlákna se zobrazí počet výskytů" );
            System.out.println( " Inicializace druhého vlákna k provedení vyhledání" );
            System.out.println( " Spánek 0,5 sekundy" );
            System.out.println( " Zrušit vyhledávání" );
            System.out.println( " Po dokončení druhého vlákna se zobrazí počet výskytů" );
            System.out.println( "" );
            System.out.println( "Zajistěte, aby byla vybrána složka, která obsahuje kritérium s názvem Date." );
            System.out.println( "Zajistěte, aby složka obsahovala hodně výskytů a aby soubor arswww.ini neomezoval" );
            System.out.println( "příliš počet výskytů, které mohou být vráceny." );
            System.out.println( "" );
            System.out.println( "-----" );
            System.out.println( "" );

            //-----
            // Přihlášení k zadanému serveru
            //-----
            odServer = new ODServer( );
            odServer.initialize( argv[4], "TcCancelSearch.java" );

            System.out.println( "Přihlášení k " + argv[0] + "..." );
            if ( argv.length == 5 )
                odServer.logon( argv[0], argv[1], argv[2] );
            else
                if ( argv.length == 6 )
                    odServer.logon( argv[0], argv[1], argv[2], ODConstant.CONNECT_TYPE_LOCAL, 0, argv[5] );

            //-----
            // Otevření zadané složky a zobrazení jejího jména a popisu
            //-----
            System.out.println( "Otevření " + argv[3] + "..." );
        }
    }
}
```

```

odFolder = odServer.openFolder( argv[3] );
odCrit = odFolder.getCriteria( "Date" );
odCrit.setOperand( ODConstant.OPBetween );
odCrit.setSearchValues( "01/01/70", "01/01/01" );

//-----
// Spuštění vyhledávání na jiném vláknu, krátký spánek, probuzení a zrušení hledání
//-----
System.out.println( "Hlavní vlákno inicializuje vyhledávání (nebude se pokoušet o zrušení)..." );
search_thread = new TestThread( odFolder );
search_thread.start( );
search_thread.join( );

System.out.println( "Hlavní vlákno inicializuje vyhledávání (bude se pokoušet o zrušení)..." );
search_thread = new TestThread( odFolder );
search_thread.start( );
System.out.println( "Hlavní vlákno spí 0,5 sekund ..." );
( Thread.currentThread( ) ).sleep( 500 );
System.out.println( "Hlavní vlákno se pokouší zrušit vyhledávání..." );
odServer.cancel( );
System.out.println( "Hlavní vlákno se vrátilo z pokusu o zrušení" );
search_thread.join( );

//-----
// Čištění
//-----
odFolder.close( );
odServer.logoff( );
odServer.terminate( );
System.out.println( "" );
System.out.println( "-----" );
System.out.println( "" );
System.out.println( "Testovací případ dokončen - ověřte, zda druhé vyhledávání, " );
System.out.println( " které bylo zrušeno, přineslo méně výsledků než první " );
System.out.println( "" );
}

catch ( ODEException e )
{
    System.out.println( "ODEException: " + e );
    System.out.println( " id = " + e.getErrorId( ) );
    System.out.println( " msg = " + e.getErrorMsg( ) );
    e.printStackTrace( );
}

catch ( Exception e2 )
{
    System.out.println( "výjimka: " + e2 );
    e2.printStackTrace( );
}
}
}

```

Výpis kritérií hledání

Následující příklad demonstruje, jak používat metody `ODCriteria` pro výpis kritérií hledání pro danou složku. Pro každé vyhledávací pole tento příklad uvádí jméno vyhledávacího pole, předvolený operátor, operátory, které jsou platné pro pole, typ pole a předvolené hodnoty hledání. Předvolené hodnoty jsou uváděny metodami `ODCriteria.getSearchValues` a `ODCriteria.getValues`. Pevné hodnoty hledání jsou uváděny pro všechna vyhledávací pole, která jsou definována jako `FixedChoice` nebo `Segment`.

Tento příklad ukazuje použití těchto metod `ODCriteria`:

- `setOperand`
- `getValidOperands`
- `getType`
- `getValues`
- `setSearchValues`
- `getFixedValues`

V tomto příkladu jsou předvedeny tyto metody `ODServer`:

- `initialize`
- `logon`
- `openFolder`
- `logoff`
- `terminate`

Tento příklad ukazuje použití těchto metod `ODFolder`:

- `getCriteria`
- `close`

V tomto příkladu jsou použity tyto runtime parametry:

- Jméno serveru
- ID uživatele
- Heslo
- Jméno složky
- Konfigurační adresář (umístění souboru arswwww.ini)
- (volitelné) Adresář lokálního serveru

Příklad přístupu ke kritériím vyhledávání:

```
//*****
import java.util.*;
import java.io.*;
import com.ibm.edms.od.*;

public class TcListCriteria
{
    public static void main ( String argv[] )
    {
        ODServer odServer;
        ODFolder odFolder;
        ODCriteria odCrit;
        Enumeration crit_enum;
        Vector value_vec;
        String[] search_values, fixed_values;
        int[] valid_oprs;
        int j, opr;
        char ffield_type;

        //-----
        // Je-li příliš málo parametrů, zobrazit syntaxi a odejít
        //-----
        if ( argv.length < 5 )
        {
            System.out.println( "usage: java TcListCriteria <server> <ID uživatele> <heslo> <složka> <konfigurační adresář>
            [<adresář lokálního serveru>]" );
            return;
        }

        try
        {
            //-----
            // Vysvětlení
            //-----
            System.out.println( "Tento testovací případ by měl:" );
            System.out.println( " Přihlášení k zadanému serveru" );
            System.out.println( " Otevření zadané složky" );
            System.out.println( " Zobrazení jména a popisu složky" );
            System.out.println( " Zobrazení počtu kritérií složky" );
            System.out.println( " Pro každé kritérium se zobrazí" );
            System.out.println( " Jméno" );
            System.out.println( " Předvolený operátor" );
            System.out.println( " Platné operátory" );
            System.out.println( " Typ pole" );
            System.out.println( " Předvolené hodnoty (metodou ODCrit.getSearchValues)" );
            System.out.println( " Předvolené hodnoty (metodou ODCrit.getValues)" );
            System.out.println( " Pevné hodnoty (pouze pro kritéria FixedChoice a Segment)" );
            System.out.println( "" );
            System.out.println( "Zajistěte, aby žádný z operátorů neuváděl 'Neznámý operátor,' " );
            System.out.println( "aby žádné typy polí neuváděly 'Neznámý typ,' aby" );
            System.out.println( "předvolené hodnoty byly stejné pro každou metodu, a aby všechny" );
            System.out.println( "informace byly stejné jako ty, které jsou zobrazeny klientem Windows." );
            System.out.println( "" );
            System.out.println( "-----" );
            System.out.println( "" );

            //-----
            // Přihlášení k zadanému serveru
            //-----
            odServer = new ODServer( );
            odServer.initialize( argv[4], "TcListCriteria.java" );

            System.out.println( "Přihlášení k " + argv[0] + "..." );
            if ( argv.length == 5 )
                odServer.logon( argv[0], argv[1], argv[2] );
            else
                if ( argv.length == 6 )
                    odServer.logon( argv[0], argv[1], argv[2], ODCConstant.CONNECT_TYPE_LOCAL, 0, argv[5] );

            //-----
            // Otevření zadané složky a zobrazení jejího jména a popisu
            //-----
            System.out.println( "Otevření složky " + argv[3] + "..." );
            odFolder = odServer.openFolder( argv[3] );
            System.out.println( "Jméno=" + odFolder.getName( ) + " Popis=" + odFolder.getDescription( ) + "" );
            System.out.println( "Existují " + odFolder.getNumCriteria( ) + " kritéria:" );

            //-----
            // Pro každé kritérium složky,
            //-----
            for ( crit_enum = odFolder.getCriteria( ); crit_enum.hasMoreElements( ); )
            {
                //-----
                // Zobrazení jména kritéria
                //-----
                System.out.println( "" );
            }
        }
    }
}
```

```

odCrit = (ODCriteria)crit_enum.nextElement( );
System.out.println( odCrit.getName( ) );

//-----
// Zobrazení předvoleného operátoru
//-----
opr = odCrit.getOperand( );
System.out.println( " Předvolený operátor: " );
System.out.println( " " + getOperatorName( opr ) );

//-----
// Zobrazení platných operátorů
//-----
valid_oprs = odCrit.getValidOperands( );
System.out.println( " Platné operátory:" );
for ( j = 0; j < valid_oprs.length; j++ )
    System.out.println( " " + getOperatorName( valid_oprs[j] ) );

//-----
// Zobrazení typu pole
//-----
field_type = odCrit.getType( );
System.out.println( " Type:" );
System.out.println( " " + getTypeName( field_type ) );

//-----
// Zobrazení předvolené hodnoty (hodnot) pomocí ODCrit.getValues( )
//-----
value_vec = odCrit.getValues( );
System.out.println( " Předvolená hodnota (hodnoty) (ODCrit.getValues method):" );
System.out.println( " " + value_vec.elementAt( 0 ) + " " );
System.out.println( " " + value_vec.elementAt( 1 ) + " " );

//-----
// Zobrazení předvolené hodnoty (hodnot) pomocí ODCrit.getSearchValues( )
//-----
search_values = odCrit.getSearchValues( );
System.out.println( " Předvolené hodnoty (ODCrit.getSearchValues method):" );
for ( j = 0; j < search_values.length; j++ )
    System.out.println( " " + search_values[j] + " " );

//-----
// Zobrazení pevných voleb
//-----
switch ( field_type )
{
    case ODConstant.InputTypeChoice:
    case ODConstant.InputTypeSegment:
        fixed_values = odCrit.getFixedValues( );
        System.out.println( " Pevné hodnoty (pouze pro typy polí FixedChoice a Segment):" );
        for ( j = 0; j < fixed_values.length; j++ )
            System.out.println( " " + fixed_values[j] + " " );
        break;
}

//-----
// Čištění
//-----
odFolder.close( );
odServer.logoff( );
odServer.terminate( );
System.out.println( " " );
System.out.println( "-----" );
System.out.println( " " );
System.out.println( "Testovací případ dokončen - analyzujte a porovnejte výsledky s " );
System.out.println( " klientem Windows, je-li to zapotřebí" );
System.out.println( " " );
}

catch ( ODEException e )
{
    System.out.println( "ODEException: " + e );
    System.out.println( " id = " + e.getErrorId( ) );
    System.out.println( " msg = " + e.getErrorMsg( ) );
    e.printStackTrace( );
}

catch ( Exception e2 )
{
    System.out.println( "výjimka: " + e2 );
    e2.printStackTrace( );
}

static String getOperatorName( int oper )
{
    String str;

    switch( oper )
    {
        case ODConstant.OPEqual:
            str = "Equal";
            break;
        case ODConstant.OPNotEqual:
            str = "Not Equal";
            break;
        case ODConstant.OPLessThan:
            str = "Less Than";
            break;
        case ODConstant.OPLessThanEqual:
            str = "Less Than or Equal";
            break;
    }
}

```

```

        case ODConstant.OPGreaterThan:
            str = "Greater Than";
            break;
        case ODConstant.OPGreaterThanEqual:
            str = "Greater Than or Equal";
            break;
        case ODConstant.OPIn:
            str = "In";
            break;
        case ODConstant.OPNotIn:
            str = "Not In";
            break;
        case ODConstant.OLike:
            str = "Like";
            break;
        case ODConstant.OPNotLike:
            str = "Not Like";
            break;
        case ODConstant.OPBetween:
            str = "Between";
            break;
        case ODConstant.OPNotBetween:
            str = "Not Between";
            break;
        default:
            str = "*** Neznámý operátor";
            break;
    }

    return str;
}

static String getTypeName( char type )
{
    String str;

    switch( type )
    {
        case ODConstant.InputTypeNormal:
            str = "Normal";
            break;
        case ODConstant.InputTypeTextSearch:
            str = "TextSearch";
            break;
        case ODConstant.InputTypeNoteTextSearch:
            str = "NoteTextSearch";
            break;
        case ODConstant.InputTypeNoteColor:
            str = "NoteColor";
            break;
        case ODConstant.InputTypeChoice:
            str = "FixedChoice";
            break;
        case ODConstant.InputTypeSegment:
            str = "Segment";
            break;
        default:
            str = "*** Neznámý typ";
            break;
    }

    return str;
}
}

```

Výpis složek a informací o složkách

V následujícím příkladu jsou použity metody `ODServer` pro tisk řádku uvádějícího počet složek na zadaném serveru, které mohou být prohledávány zadaným ID uživatele. Příklad tiskne jeden řádek pro každou složku a uvádí jméno a popis složky.

V tomto příkladu jsou předvedeny tyto metody `ODServer`:

- `initialize`
- `logon`
- `getNumFolders`
- `getFolderNames`
- `getFolderDescription`
- `logoff`
- `terminate`

V tomto příkladu jsou použity tyto runtime parametry:

- Jméno serveru
- ID uživatele
- Heslo
- Konfigurační adresář (umístění souboru `arswww.ini`)

- (volitelné) Adresář lokálního serveru

Příklad výpisu složek a informací o složkách:

```

/*****
import java.util.*;
import java.io.*;
import com.ibm.edms.od.*;

public class TcListFolders
{
    public static void main ( String argv[] )
    {
        ODServer    odServer;
        Enumeration folders_enum;
        String folder_name, folder_desc;
        int num_folders;

        //-----
        // Je-li příliš málo parametrů, zobrazit syntaxi a odejít
        //-----
        if ( argv.length < 4 )
        {
            System.out.println( "použití: java TcListFolders <server> <ID uživatele> <heslo>
            <konfigurační adresář> [<adresář lokálního serveru>]" );
            return;
        }

        try
        {
            //-----
            // Vysvětlení
            //-----
            System.out.println( "Tento testovací případ by měl:" );
            System.out.println( "  Zobrazení řádku uvádějícího počet složek na serveru, které jsou k dispozici pro ID uživ." );
            System.out.println( "  Zobrazení jednoho řádku pro každou složku uvádějícího jméno a popis" );
            System.out.println( "  " );
            System.out.println( "Informace by měly být stejné jako ty, které jsou zobrazeny klientem Windows" );
            System.out.println( "(se zaškrtnutým tlačítkem 'All', je-li k dispozici), ale pořadí složek" );
            System.out.println( "se může lišit v závislosti na zadaném serveru" );
            System.out.println( "  " );
            System.out.println( "-----" );
            System.out.println( "  " );

            //-----
            // Přihlášení k zadanému serveru
            //-----
            odServer = new ODServer( );
            odServer.initialize( argv[3], "TcListFolders.java" );

            System.out.println( "Přihlášení k " + argv[0] + "..." );
            if ( argv.length == 4 )
                odServer.logon( argv[0], argv[1], argv[2] );
            else
                if ( argv.length == 5 )
                    odServer.logon( argv[0], argv[1], argv[2], ODConstant.CONNECT_TYPE_LOCAL, 0, argv[4] );

            //-----
            // Zobrazte počet dostupných složek.
            //-----
            num_folders = odServer.getNumFolders( );
            System.out.println( "  " );
            System.out.println( "Existuje " + num_folders + " složek dostupných pro " + argv[1] + " na " + argv[0] + ":" );

            //-----
            // Zobrazení jmen a popisů složek
            //-----
            for ( folders_enum = odServer.getFolderNames( ); folders_enum.hasMoreElements( ); )
            {
                folder_name = (String)folders_enum.nextElement( );
                folder_desc = odServer.getFolderDescription( folder_name );
                System.out.println( "  " + folder_name + " --- " + folder_desc );
            }

            //-----
            // Čištění
            //-----
            odServer.logoff( );
            odServer.terminate( );
            System.out.println( "  " );
            System.out.println( "-----" );
            System.out.println( "  " );
            System.out.println( "Testovací případ dokončen - porovnejte výsledky s klientem Windows, jestliže je to zapotřebí" );
            System.out.println( "  " );
        }

        catch ( ODException e )
        {
            System.out.println( "ODException: " + e );
            System.out.println( "  id = " + e.getErrorId( ) );
            System.out.println( "  msg = " + e.getErrorMsg( ) );
            e.printStackTrace( );
        }
    }
}

```

```

        catch ( Exception e2 )
        {
            System.out.println( "výjimka: " + e2 );
            e2.printStackTrace( );
        }
    }
}

```

Zobrazení seznamu dokumentů

Následující příklad uvádí metody ODFolder a ODHit pro vyhledání složky pomocí předvolených vyhledávacích kritérií, pro tisk počtu dokumentů, které odpovídají dotazu, a pro výpis dokumentů, které odpovídají dotazu.

Tento příklad ukazuje použití těchto metod ODFolder:

- getName
- getDisplayOrder
- search
- close

V tomto příkladu jsou předvedeny tyto metody ODHit:

- getDisplayValue

V tomto příkladu jsou také předvedeny tyto metody ODServer:

- initialize
- logon
- openFolder
- logoff
- terminate

V tomto příkladu jsou použity tyto runtime parametry:

- Jméno serveru
- ID uživatele
- Heslo
- Jméno složky
- Konfigurační adresář (umístění souboru arswwww.ini)

Příklad zobrazení seznamu dokumentů:

```

//*****
import java.util.*;
import java.io.*;
import com.ibm.edms.od.*;

public class TcSortedHitlist
{
    public static void main ( String argv[] )
    {
        ODServer odServer;
        ODFolder odFolder;
        ODHit odHit;
        Vector hits;
        String[] display_crit;
        String server, userid, password, folder, value;
        int j, k;

        //-----
        // Je-li příliš málo parametrů, zobrazit syntaxi a odejít
        //-----
        if ( argv.length < 5 )
        {
            System.out.println( "použití: java TcSortedHitlist <server> <ID uživatele> <heslo> <složka>
<konfigurační adresář>" );
            return;
        }
    }

    try
    {
        //-----
        // Vysvětlení
        //-----

```

```

System.out.println( "Tento testovací případ by měl:" );
System.out.println( " Přihlášení k zadanému serveru" );
System.out.println( " Otevření zadané složky" );
System.out.println( " Vyhledání složky pomocí předvolených kritérií" );
System.out.println( " Zobrazení hledací zprávy (pokud se požaduje)" );
System.out.println( " Zobrazení počtu výskytů" );
System.out.println( " Zobrazení seznamu výskytů" );
System.out.println( "" );
System.out.println( "-----" );
System.out.println( "" );

//-----
// Přihlášení k serveru
//-----
server = argv[0];
userid = argv[1];
password = argv[2];
folder = argv[3];
odServer = new ODServer( );
odServer.initialize( argv[4], "TcSortedHitlist.java" );
System.out.println( "Přihlášení k " + server + " jako " + userid + "/" + password + "..." );
odServer.logon( server, userid, password );

//-----
// Otevření a vyhledání složky
//-----
System.out.println( "Otevírání " + složka + "..." );
odFolder = odServer.openFolder( folder );
System.out.println( "Vyhledání složky s předvolenými kritérii..." );
hits = odFolder.search( );
System.out.println( " Počet výskytů: " + hits.size( ) );

//-----
// Zobrazení výskytů
//-----
if ( hits != null && hits.size( ) > 0 )
{
    display_crit = odFolder.getDisplayOrder( );
    value = " ";
    for( j = 0; j < display_crit.length; j++ )
        value = value + display_crit[j] + " ";
    System.out.println( value );
    for ( j = 0; j < hits.size( ); j++ )
    {
        odHit = (ODHit)hits.elementAt( j );
        value = " ";
        for ( k = 0; k < display_crit.length; k++ )
            value = value + odHit.getDisplayValue( display_crit[k] ) + " ";
        System.out.println( value );
    }
}

//-----
// Čištění
//-----
odFolder.close( );
odServer.logoff( );
odServer.terminate( );
System.out.println( "" );
System.out.println( "-----" );
System.out.println( "" );
System.out.println( "Testovací případ dokončen - ověřte, zda pořadí výskytů" );
System.out.println( " je stejné jako je uvedeno v klientovi Windows" );
System.out.println( "" );
}

catch ( ODEException e )
{
    System.out.println( "ODEException: " + e );
    System.out.println( " id = " + e.getErrorId( ) );
    System.out.println( " msg = " + e.getErrorMsg( ) );
    e.printStackTrace( );
}

catch ( Exception e2 )
{
    System.out.println( "výjimka: " + e2 );
    e2.printStackTrace( );
}
}
}

```

Načtení dokumentu

V následujícím příkladu jsou předvedeny tři rozdílné metody načítání dokumentu:

- ODServer
- ODFolder
- ODHit

Tento příklad se přihlásí k zadanému serveru, otevře zadanou složku, složku prohledá s použitím předvolených kritérií, zobrazí počet výskytů, načte data pro první výskyt pomocí ODHit.retrieve, načte data pro první výskyt pomocí ODServer.retrieve a načte data pro první výskyt pomocí ODFolder.retrieve. Tento příklad uvádí délku dat načtených z každé metody, porovnává délky a data načtená z každé metody a zobrazuje výsledek z porovnání.

V tomto příkladu jsou předvedeny tyto metody ODServer:

- initialize
- logon
- openFolder
- retrieve
- logoff
- terminate

Tento příklad ukazuje použití těchto metod ODFolder:

- search
- retrieve
- close

V tomto příkladu jsou předvedeny tyto metody ODHit:

- getDocId
- retrieve

V tomto příkladu jsou použity tyto runtime parametry:

- Jméno serveru
- ID uživatele
- Heslo
- Jméno složky
- Konfigurační adresář (umístění souboru arswwww.ini)
- (volitelné) Adresář lokálního serveru

Příklad načtení dokumentu:

```
//*****
import java.util.*;
import java.io.*;
import com.ibm.edms.od.*;

public class TcRetrieve
{
    public static void main ( String argv[] )
    {
        ODServer odServer;
        ODFolder odFolder;
        ODHit odHit;
        TcCallback callback;
        Vector hits;
        Vector hit_to_retrieve;
        byte[] data_from_hit;
        byte[] data_from_server;
        byte[] data_from_folder;
        int j;

        //-----
        // Je-li příliš málo parametrů, zobrazit syntaxi a odejít
        //-----
        if ( argv.length < 5 )
        {
            System.out.println( "použití: java TcRetrieve <server> <ID uživatele> <heslo> <složka> <konfigurační adresář>
            [<adresář lokálního serveru>]" );
            return;
        }
    }

    try
```

```

{
//-----
// Vysvětlení
//-----
System.out.println( "Tento testovací případ by měl:" );
System.out.println( " Přihlášení k zadanému serveru" );
System.out.println( " Otevření zadané složky" );
System.out.println( " Vyhledání složky pomocí předvolených kritérií" );
System.out.println( " Zobrazení počtu výskytů" );
System.out.println( " Načtení dat prvního výskytu pomocí ODHit.retrieve" );
System.out.println( " Načtení dat prvního výskytu pomocí ODServer.retrieve" );
System.out.println( " Načtení dat prvního výskytu pomocí ODFolder.retrieve" );
System.out.println( " Zobrazení délky dat přijatý z každé metody" );
System.out.println( " Porovnání délek a dat přijatých z každé metody" );
System.out.println( " Zobrazení výsledku porovnání" );
System.out.println( "" );
System.out.println( "-----" );
System.out.println( "" );

//-----
// Přihlášení k zadanému serveru
//-----
odServer = new ODServer( );
odServer.initialize( argv[4], "TcRetrieve.java" );
System.out.println( "Přihlášení k " + argv[0] + "..." );
if ( argv.length == 5 )
odServer.logon( argv[0], argv[1], argv[2] );
else
odServer.logon( argv[0], argv[1], argv[2], ODConstant.CONNECT_TYPE_LOCAL, 0, argv[5] );

//-----
// Otevření zadané složky a vyhledání s požadovanými kritérii
//-----
System.out.println( "Otevření " + argv[3] + " složka..." );
odFolder = odServer.openFolder( argv[3] );
System.out.println( "Vyhledání s předvolenými kritérii..." );
hits = odFolder.search( );
System.out.println( "Počet výskytů: " + hits.size( ) );

//-----
// Proveďte některá načtení a porovnání
//-----
if ( hits.size( ) > 0 )
{
odHit = (ODHit)hits.elementAt( 0 );
System.out.println( "Načtení dat z prvního výskytu pomocí ODHit.retrieve..." );
data_from_hit = odHit.retrieve( "" );
System.out.println( "Načtení dat z prvního výskytu pomocí ODServer.retrieve..." );
data_from_server = odServer.retrieve( odHit.getDocId( ), argv[3], "" );
hit_to_retrieve = new Vector( );
hit_to_retrieve.addElement( odHit );
System.out.println( "Načtení dat z prvního výskytu pomocí ODFolder.retrieve (používá metodu callback)..." );
callback = new TcCallback( );
odFolder.retrieve( hit_to_retrieve, callback );
data_from_folder = callback.getData( );
System.out.println( "Délka dat z:" );
System.out.println( " ODHit.retrieve=" + data_from_hit.length );
System.out.println( " ODServer.retrieve=" + data_from_server.length );
System.out.println( " ODFolder.retrieve=" + data_from_folder.length );
if ( data_from_hit.length == data_from_server.length )
{
for ( j = 0; j < data_from_hit.length; j++ )
{
if ( data_from_hit[j] != data_from_server[j] )
break;
}
if ( j == data_from_hit.length )
{
System.out.println( "ODHit vs. ODServer: Délka a obsah datové shody" );
if ( data_from_hit.length == data_from_folder.length )
{
for ( j = 0; j < data_from_folder.length; j++ )
{
if ( data_from_hit[j] != data_from_folder[j] )
break;
}
if ( j == data_from_folder.length )
System.out.println( "ODHit vs. ODFolder: Délka a obsah datových shod" );
else
{
System.out.println( "*** ODHit vs. ODFolder: Neshoda dat na odstupu " + j );
System.out.println( " data ODHit jsou " + data_from_hit[j] );
System.out.println( " data ODFolder jsou " + data_from_folder[j] );
}
}
else
System.out.println( "*** ODHit vs. ODFolder: Neshoda délky" );
}
else
{
System.out.println( "*** ODHit vs. ODServer: Neshoda dat na odstupu " + j );
System.out.println( " data ODHit jsou " + data_from_hit[j] );
System.out.println( " data ODServer jsou " + data_from_server[j] );
}
}
else
System.out.println( "*** ODHit vs. ODServer: Neshoda délky" );
}
else
System.out.println( "Není žádný dokument k načtení" );
}
}

```



```

//-----
// Čištění
//-----
odFolder.close( );
odServer.logoff( );
odServer.terminate( );
System.out.println( "" );
System.out.println( "-----" );
System.out.println( "" );
System.out.println( "Testovací případ dokončen - analyzujte výsledek porovnání" );
System.out.println( "" );
System.out.println( "Jestliže soubor arswww.ini uvádí 'native' jako datový typ, všechny" );
System.out.println( "délka a data by se měly shodovat; jinak lze očekávat rozdíly." );
System.out.println( "" );
}

catch ( ODEException e )
{
    System.out.println( "ODEException: " + e );
    System.out.println( " id = " + e.getErrorId( ) );
    System.out.println( " msg = " + e.getErrorMsg( ) );
    e.printStackTrace( );
}

catch ( Exception e2 )
{
    System.out.println( "výjimka: " + e2 );
    e2.printStackTrace( );
}
}
}

```

Následující příklad používá metody ODCallback pro hromadné načtení dokumentových dat.

```

//*****
import java.util.*;
import java.io.*;
import com.ibm.edms.od.*;

public class TcCallback extends ODCallback
{
    byte[] data_from_folder;
    boolean init = true;

    TcCallback( )
    {
    }

    public void HitHandleCallback( int hit, int off, int len )
    {
    }

    public boolean HitCallback( String docid, char type, String[] values )
        throws Exception
    {
        return true;
    }

    public boolean DataCallback( byte[] data )
    {
        byte[] temp;
        int j, k;

        //-----
        // Jestliže byl přijat první datový blok, inicializovat zásobník; jinak připojit
        // nová data ke dříve přijatým datům.
        //-----
        if ( init )
        {
            data_from_folder = data;
            init = false;
        }
        else
        {
            temp = new byte[ data_from_folder.length + data.length ];
            for ( j = 0; j < data_from_folder.length; j++ )
                temp[j] = data_from_folder[j];
            k = data_from_folder.length;
            for ( j = 0; j < data.length; j++ )
                temp[k++] = data[j];
            data_from_folder = temp;
        }

        return true;
    }
}

```

```

public byte[] getData( )
{
    return data_from_folder;
}
}

```

Tisk dokumentu

V následujícím příkladu jsou použity metody ODServer a ODFolder pro výpis tiskáren, které jsou k dispozici na serveru a pro tisk dokumentu na zadané serverové tiskárně. Tento příklad také používá metody ODServer pro přípravu k přihlášení, otevření zadané složky a odhlášení.

V tomto příkladu jsou předvedeny tyto metody ODServer:

- initialize
- logon
- openFolder
- getServerPrinters
- logoff
- terminate

Tento příklad ukazuje použití těchto metod ODFolder:

- search
- printDocs
- close

V tomto příkladu jsou použity tyto runtime parametry:

- Jméno serveru
- ID uživatele
- Heslo
- Jméno složky
- Jméno tiskárny
- Konfigurační adresář (umístění souboru arswwww.ini)
- (volitelné) Adresář lokálního serveru

Příklad tisku dokumentu:

```

//*****
import java.util.*;
import java.io.*;
import com.ibm.edms.od.*;

public class TcPrintHit
{
    public static void main ( String argv[] )
    {
        ODServer odServer;
        ODFolder odFolder;
        ODHit odHit;
        Vector hits, hit_to_print;
        String [] printers;
        String printer_name;
        boolean match;
        int j;

        //-----
        // Je-li příliš málo parametrů, zobrazit syntaxi a odejít
        //-----
        if ( argv.length < 6 )
        {
            System.out.println( "použití: java TcPrintHit <server> <ID uživatele> <heslo> <složka> <tiskárna> <konfigurační adresář>
            [<adresář lokálního serveru>]" );
            return;
        }

        try
        {
            //-----
            // Vysvětlení
            //-----
            System.out.println( "Tento testovací případ by měl:" );
            System.out.println( " Přihlášení k zadanému serveru" );
            System.out.println( " Zobrazení seznamu tiskáren dostupných na serveru" );
            System.out.println( " Otevření zadané složky" );
            System.out.println( " Vyhledání složky pomocí předvolených kritérií" );
            System.out.println( " Zobrazení počtu výskytů" );
            System.out.println( " Tisk prvního výskytu na speciální serverové tiskárně" );
            System.out.println( "" );
        }
    }
}

```

```

System.out.println( "-----" );
System.out.println( "" );

//-----
// Přihlášení k zadanému serveru
//-----
odServer = new ODServer( );
odServer.initialize( argv[5], "TcPrintHit.java" );
System.out.println( "Přihlášení k " + argv[0] + "..." );
if ( argv.length == 6 )
    odServer.logon( argv[0], argv[1], argv[2] );
else
    odServer.logon( argv[0], argv[1], argv[2], ODConstant.CONNECT_TYPE_LOCAL, 0, argv[6] );

//-----
// Jestliže jsou na serveru k dispozici nějaké tiskárny
//-----
System.out.println( "Načtení seznamu serverových tiskáren..." );
printer_name = argv[4];
printers = odServer.getServerPrinters( );
if ( printers.length > 0 )
{
    //-----
    // Seznam dostupných serverových tiskáren
    //-----
    System.out.println( "Existuje " + printers.length + " tiskáren dostupných na serveru:" );
    match = false;
    for( j = 0; j < printers.length; j++ )
    {
        System.out.println( " " + printers[j] );
        if ( printers[j].equals( printer_name ) )
            match = true;
    }

    if ( match )
    {
        //-----
        // Otevření zadané složky a vyhledání s požadovanými kritérii
        //-----
        System.out.println( "Otevírání " + argv[3] + " složka..." );
        odFolder = odServer.openFolder( argv[3] );
        System.out.println( "Vyhledání s předvolenými kritérii..." );
        hits = odFolder.search( );
        System.out.println( " Počet výskytů: " + hits.size( ) );

        //-----
        // Tisk prvního výskytu na zadanou serverovou tiskárnu
        //-----
        if ( hits.size( ) > 0 )
        {
            hit_to_print = new Vector( );
            odHit = (ODHit)hits.elementAt( 0 );
            hit_to_print.addElement( odHit );
            System.out.println( "Tisk prvního výskytu na " + printer_name + "..." );
            odFolder.printDocs( hit_to_print, printer_name );
        }
        else
            System.out.println( "Není žádný dokument k tisku" );

        odFolder.close( );
    }
    else
        System.out.println( "Zadaná tiskárna ( " + printer_name + " ) není na tomto serveru k dispozici" );
}
else
    System.out.println( "Na tomto serveru nejsou k dispozici žádné tiskárny" );

//-----
// Čištění
//-----
odServer.logoff( );
odServer.terminate( );
System.out.println( "" );
System.out.println( "-----" );
System.out.println( "" );
System.out.println( "Testovací případ dokončen - analyzujte výsledky" );
System.out.println( "" );
}

catch ( ODEException e )
{
    System.out.println( "ODEException: " + e );
    System.out.println( " id = " + e.getErrorId( ) );
    System.out.println( " msg = " + e.getErrorMsg( ) );
    e.printStackTrace( );
}

catch ( Exception e2 )
{
    System.out.println( "výjimka: " + e2 );
    e2.printStackTrace( );
}
}

```

Výpis informací o poznámkách

V následujícím příkladu jsou použity metody ODNote pro výpis podrobných informací o poznámce. Tento příklad se přihlásí k zadanému serveru, otevře zadanou složku, složku prohledá s použitím předvolených kritérií, zobrazuje počet výskytů, zobrazuje počet poznámek přidružených k prvnímu dokumentu a zobrazuje podrobné informace o každé

poznámce, která je připojena k tomuto dokumentu. Informace také zahrnují polohu poznámky na straně dokumentu, barvu pozadí, datum a čas, kdy byla poznámka připojena k dokumentu, ID uživatele, který vytvořil poznámku a jiné atributy.

Tento příklad ukazuje použití těchto metod ODNNote:

- getColor
- getDateTime
- getGroupName
- getOffsetX
- getOffsetY
- getPageNum
- getText
- getUserid
- isOkToCopy
- isPublic

V tomto příkladu jsou také předvedeny tyto metody ODServer:

- initialize
- logon
- openFolder
- logoff
- terminate

Tento příklad také ukazuje použití těchto metod ODFolder:

- search
- close

V tomto příkladu jsou také předvedeny tyto metody ODHit:

- getNotes

V tomto příkladu jsou použity tyto runtime parametry:

- Jméno serveru
- ID uživatele
- Heslo
- Jméno složky
- Konfigurační adresář (umístění souboru arswwww.ini)
- (volitelné) Adresář lokálního serveru

Příklad výpisu informací o poznámkách:

```
//*****
import java.util.*;
import java.io.*;
import com.ibm.edms.od.*;

public class TcListNotes
{
    public static void main ( String argv[] )
    {
        ODServer odServer;
        ODFolder odFolder;
        ODHit odHit;
        ODNNote odNote;
        Vector hits, notes;
        int j;

        //-----
        // Je-li příliš málo parametrů, zobrazit syntaxi a odejít
        //-----
        if ( argv.length < 5 )
        {
            System.out.println( "použití: java TcListNotes <server> <ID uživatele> <heslo> <složka> <konfigurační adresář>
            [<adresář lokálního serveru>" );
            return;
        }

        try
        {
            //-----
            // Vysvětlení
            //-----
            System.out.println( "Tento testovací případ by měl:" );
        }
    }
}
```

```

System.out.println( " Přihlášení k zadanému serveru" );
System.out.println( " Otevření zadané složky" );
System.out.println( " Vyhledání složky pomocí předvolených kritérií" );
System.out.println( " Zobrazení počtu výskytů" );
System.out.println( " Zobrazení počtu poznámek přidružených k prvnímu výskytu" );
System.out.println( " Zobrazení informací o každé poznámce" );
System.out.println( "" );
System.out.println( "-----" );
System.out.println( "" );

//-----
// Přihlášení k zadanému serveru
//-----
odServer = new ODServer( );
odServer.initialize( argv[4], "TcListNotes.java" );
System.out.println( "Přihlášení k " + argv[0] + "..." );
if ( argv.length == 5 )
    odServer.logon( argv[0], argv[1], argv[2] );
else
    odServer.logon( argv[0], argv[1], argv[2], ODConstant.CONNECT_TYPE_LOCAL, 0, argv[6] );

//-----
// Otevření zadané složky a vyhledání s požadovanými kritérii
//-----
System.out.println( "Otvírání " + argv[3] + " složka..." );
odFolder = odServer.openFolder( argv[3] );
System.out.println( "Vyhledání s předvolenými kritérii..." );
hits = odFolder.search( );
System.out.println( " Počet výskytů: " + hits.size( ) );

//-----
// Vypsání informací o každé poznámce k prvnímu výskytu
//-----
if ( hits.size( ) > 0 )
{
    odHit = (ODHit)hits.elementAt( 0 );
    notes = odHit.getNotes( );
    System.out.println( " Existuje " + notes.size( ) + " poznámek pro první výskyt" );
    for ( j = 0; j < notes.size( ); j++ )
    {
        odNote = (ODNote)notes.elementAt( j );
        System.out.println( " " + (j+1) + ". Text='" + odNote.getText( ) + "'" );
        System.out.println( "   UserId=" + odNote.getUserId( ) );
        System.out.println( "   Page=" + odNote.getPageNum( ) );
        System.out.println( "   Color=" + odNote.getColor( ) );
        System.out.println( "   Date=" + odNote.getDateTime( ) );
        System.out.println( "   Group=" + odNote.getGroupName( ) );
        System.out.println( "   Offset=" + odNote.getOffsetX( ) + ", " + odNote.getOffsetY( ) + " " );
        System.out.println( "   OkToCopy=" + odNote.isOkToCopy( ) );
        System.out.println( "   Public=" + odNote.isPublic( ) );
    }
}
else
    System.out.println( "Neexistuje dokument - nelze vypsát poznámky" );

//-----
// Čištění
//-----
odFolder.close( );
odServer.logoff( );
odServer.terminate( );
System.out.println( "" );
System.out.println( "-----" );
System.out.println( "" );
System.out.println( "Testovací případ dokončen - ověřte, že informace" );
System.out.println( " jsou stejné jako je uvedeno v klientovi Windows" );
System.out.println( "" );
}

catch ( ODEException e )
{
    System.out.println( "ODEException: " + e );
    System.out.println( " id = " + e.getErrorId( ) );
    System.out.println( " msg = " + e.getErrorMsg( ) );
    e.printStackTrace( );
}

catch ( Exception e2 )
{
    System.out.println( "výjimka: " + e2 );
    e2.printStackTrace( );
}
}
}

```

Přidání poznámky

Objekt třídy ODHit představuje dokument OnDemand. V následujícím příkladu jsou použity metody ODHit k zobrazení počtu poznámek přiřazených k dokumentu a přidání nové poznámky s těmito atributy:

- Zadaný text poznámky
- OkToCopy=false
- Public=false (tj. soukromá poznámka)

- Prázdné jméno skupiny

V tomto příkladu jsou předvedeny tyto metody ODHit:

- getNotes
- addNote

Tento příklad také používá metody ODServer pro přípravu k přihlášení, otevření zadané složky a odhlášení a používá metody ODFolder pro vyhledání složky, získání počtu výskytů, které odpovídají dotazu, a zavření složky. V tomto příkladu jsou předvedeny tyto metody ODServer:

- initialize
- logon
- openFolder
- logoff
- terminate

Tento příklad ukazuje použití těchto metod ODFolder:

- search
- getHits
- close

V tomto příkladu jsou použity tyto runtime parametry:

- Jméno serveru
- ID uživatele
- Heslo
- Jméno složky
- Text poznámky
- Konfigurační adresář (umístění souboru arswwww.ini)
- (volitelné) Adresář lokálního serveru

Příklad přidání anotace:

```
//*****
import java.util.*;
import java.io.*;
import com.ibm.edms.od.*;

public class TcAddNote
{
    public static void main ( String argv[] )
    {
        ODServer odServer;
        ODFolder odFolder;
        ODHit odHit;
        ODNote odNote;
        Vector hits, notes;
        int j;

        //-----
        // Je-li příliš málo parametrů, zobrazit syntaxi a odejít
        //-----
        if ( argv.length < 6 )
            System.out.println( "použití: java TcAddNote <server> <ID uživatele> <heslo> <složka> <text poznámky> <konfigurační adresář>
[<adresář lokálního serveru>] " );
        {
            return;
        }

        try
        {
            //-----
            // Vysvětlení
            //-----
            System.out.println( "Tento testovací případ by měl:" );
            System.out.println( " Přihlášení k zadanému serveru" );
            System.out.println( " Otevření zadané složky" );
            System.out.println( " Vyhledání složky pomocí předvolených kritérií" );
            System.out.println( " Zobrazení počtu výskytů" );
            System.out.println( " Zobrazení počtu poznámek přidružených k prvnímu výskytu" );
            System.out.println( " Přidat novou poznámku s těmito atributy" );
            System.out.println( " Zadaný text poznámky" );
            System.out.println( " OkToCopy=false" );
            System.out.println( " Public=false (tj. soukromá poznámka)" );
            System.out.println( " Prázdné jméno skupiny" );
            System.out.println( "" );
            System.out.println( "-----" );
            System.out.println( "" );

            //-----
            // Přihlášení k zadanému serveru
            //-----
            odServer = new ODServer( );
```

```

odServer.initialize( argv[5], "TcAddNote.java" );
System.out.println( "Přihlášení k " + argv[0] + "..." );
if ( argv.length == 6 )
    odServer.logon( argv[0], argv[1], argv[2] );
else
    odServer.logon( argv[0], argv[1], argv[2], ODCConstant.CONNECT_TYPE_LOCAL, 0, argv[6] );

//-----
// Otevření zadané složky a vyhledání s požadovanými kritérii
//-----
System.out.println( "Otevírání " + argv[3] + " složka..." );
odFolder = odServer.openFolder( argv[3] );
System.out.println( "Vyhledání s předvolenými kritérii..." );
odFolder.search( );
hits = odFolder.getHits( );
System.out.println( " Počet výskytů: " + hits.size( ) );

//-----
// Přidat novou poznámku
//-----
if ( hits.size( ) > 0 )
{
    odHit = (ODHit)hits.elementAt( 0 );
    notes = odHit.getNotes( );
    System.out.println( " Existuje " + notes.size( ) + " poznámek pro první výskyt" );

    odNote = new ODNote( );
    odNote.setText( argv[4] );
    odNote.setGroupName( "" );
    odNote.setOkToCopy( false );
    odNote.setPublic( false );

    System.out.println( " Přidání nové poznámky s:" );
    System.out.println( " Text=" + odNote.getText( ) + "" );
    System.out.println( " OkToCopy=" + odNote.isOkToCopy( ) );
    System.out.println( " Public=" + odNote.isPublic( ) );
    System.out.println( " Group=" + odNote.getGroupName( ) );

    odHit.addNote( odNote );
}
else
    System.out.println( "Žádný dokument - nelze vypsát poznámky" );

//-----
// Čištění
//-----
odFolder.close( );
odServer.logoff( );
odServer.terminate( );
System.out.println( "" );
System.out.println( "-----" );
System.out.println( "" );
System.out.println( "Testovací případ dokončen - ověřte, zda nová poznámka byla správně" );
System.out.println( " přidána tak, že ji zobrazíte klientem Windows" );
System.out.println( "" );
}

catch ( ODEException e )
{
    System.out.println( "ODEException: " + e );
    System.out.println( " id = " + e.getErrorId( ) );
    System.out.println( " msg = " + e.getErrorMsg( ) );
    e.printStackTrace( );
}

catch ( Exception e2 )
{
    System.out.println( "výjimka: " + e2 );
    e2.printStackTrace( );
}
}
}

```

Aktualizace dokumentu

Následující příklad ukazuje, jak aktualizovat dokument.

V tomto příkladu jsou použity metody `ODServer`, `ODFolder` a `ODCriteria` pro připojení k serveru se zadaným ID uživatele a heslem, otevření zadané složky, nastavení vyhledávacích hodnot pro dvě vyhledávací pole, nastavení vyhledávacího pole `Date` na hodnotu `null` a vyhledání složky. Pro dokument, který odpovídá dotazu, se pak používají metody `ODHit` pro aktualizaci jedné nebo více databázových hodnot.

V tomto příkladu jsou předvedeny tyto metody `ODServer`:

- `initialize`
- `logon`
- `openFolder`
- `logoff`
- `terminate`

Tento příklad ukazuje použití těchto metod `ODFolder`:

- getName
- getDisplayOrder
- getCriteria
- search
- closeinitialize

Tento příklad ukazuje použití těchto metod ODCriteria:

- setOperand
- setSearchValue

V tomto příkladu jsou předvedeny tyto metody ODHit:

- getDisplayValue
- update

V tomto příkladu jsou použity tyto runtime parametry:

- Jméno serveru
- ID uživatele
- Heslo
- Jméno složky
- Jméno kritéria 1
- Hledací hodnota 1
- Jméno kritéria 2
- Hledací hodnota 2
- Nová hledací hodnota pro nahrazení hledací hodnoty 2
- Konfigurační adresář (umístění souboru arswwww.ini)

Příklad aktualizace dokumentu:

```
//*****
import java.util.*;
import java.io.*;
import com.ibm.edms.od.*;

public class TcUpdate
{
    public static void main ( String argv[] )
    {
        ODServer odServer;
        ODFolder odFolder;
        ODCriteria odCrit;
        ODHit odHit;
        Hashtable hash;
        Vector hits;
        String[] display_crit;
        String line, crit1, crit2, value1, value2, new_value;
        int j;

        //-----
        // Je-li příliš málo parametrů, zobrazit syntaxi a odejít
        //-----
        if ( argv.length < 10 )
        {
            System.out.println( "použití: java TcUpdate <server> <ID uživatele> <heslo> <složka>
                                <kritérium1> <hodnota1> " +
                                "<kritérium2> <hodnota2> <nová hodnota2>" );
            return;
        }

        try
        {
            System.out.println( "Tento testovací případ by měl:" );
            System.out.println( " Přihlášení k zadanému serveru" );
            System.out.println( " Otevření zadané složky" );
            System.out.println( " Nastavení hledaných hodnot" );
            System.out.println( " Vyhledání složky" );
            System.out.println( " Pro první výskyt změňte hodnotu druhého zadaného kritéria" );
            System.out.println( " na novou hodnotu" );
            System.out.println( "" );
            System.out.println( "Pomocí klienta Windows zajistěte, aby hodnota byla změněna." );
            System.out.println( "" );
            System.out.println( "-----" );
            System.out.println( "" );

            //-----
            // Přihlášení k zadanému serveru
```



```

//-----
odServer = new ODServer( );
odServer.initialize( argv[9], "TcUpdate.java" );
System.out.println( "Přihlášení k " + argv[0] + "..." );
odServer.logon( argv[0], argv[1], argv[2] );

//-----
// Otevření zadané složky a nastavení požadovaných kritérií
//-----
crit1 = argv[4];
crit2 = argv[6];
value1 = argv[5];
value2 = argv[7];
new_value = argv[8];
System.out.println( "Otevírání " + argv[3] + " složka..." );
odFolder = odServer.openFolder( argv[3] );
odCrit = odFolder.getCriteria( crit1 );
odCrit.setOperand( ODConstant.OPEqual );
odCrit.setSearchValue( value1 );
odCrit = odFolder.getCriteria( crit2 );
odCrit.setOperand( ODConstant.OPEqual );
odCrit.setSearchValue( value2 );

//-----
// Prohledání složky
//-----
System.out.println( " Vyhledání " + crit1 + " = " + hodnota1 + " a " + crit2 + " = " + hodnota2 + "..." );
hits = odFolder.search( );

//-----
// Jestliže byl alespoň jeden výskyt
//-----
if ( hits != null && hits.size( ) > 0 )
{
//-----
// Zobrazení hodnot pro první výskyt
//-----
System.out.println( " Pro první výskyt:" );
line = " ";
display_crit = odFolder.getDisplayOrder( );
for( j = 0; j < display_crit.length; j++ )
line = line + display_crit[j] + " ";
System.out.println( line );
line = " ";
odHit = (ODHit)hits.elementAt( 0 );
for ( j = 0; j < display_crit.length; j++ )
line = line + odHit.getDisplayValue( display_crit[j] ) + " ";
System.out.println( line );

//-----
// Vytvoření transformační tabulky existujících dvojic kritérium/hodnota kromě kritéria 2
// které bude nastaveno na novou hodnotu. Aktualizace hodnoty výskytu
//-----
System.out.println( " Nahrazení " + crit2 + " = " + hodnota2 + " s " + crit2 + " = " + new_value );
hash = new Hashtable( );
for ( j = 0; j < display_crit.length; j++ )
{
if ( display_crit[j].equals( crit2 ) )
hash.put( display_crit[j], new_value );
else
hash.put( display_crit[j], odHit.getDisplayValue( display_crit[j] ) );
}
odHit.update( hash );
}
else
System.out.println( "Nebyly žádné výskyty" );

//-----
// Čištění
//-----
odFolder.close( );
odServer.logoff( );
odServer.terminate( );
System.out.println( "" );
System.out.println( "-----" );
System.out.println( "" );
System.out.println( "Testovací případ dokončen - pomocí klienta Windows" );
System.out.println( " ověřte, zda hodnota byla změněna." );
System.out.println( "" );
}

catch ( ODEException e )
{
System.out.println( "ODEException: " + e );
System.out.println( " id = " + e.getErrorId( ) );
System.out.println( " msg = " + e.getErrorMsg( ) );
e.printStackTrace( );
}

catch ( Exception e2 )
{
System.out.println( "výjimka: " + e2 );
}

```

```

        e2.printStackTrace( );
    }
}

```

Změna hesla

V následujícím příkladu je použita metoda `ODServer changePassword` pro změnu zadaného hesla uživatele na nové heslo. Tento příklad také používá metody `ODServer` pro přípravu k přihlášení a odhlášení.

V tomto příkladu jsou předvedeny tyto metody `ODServer`:

- initialize
- logon
- changePassword
- logoff
- terminate

V tomto příkladu jsou použity tyto runtime parametry:

- Jméno serveru
- ID uživatele
- Heslo
- Nové heslo
- Konfigurační adresář (umístění souboru `arswww.ini`)
- (volitelné) Adresář lokálního serveru

Příklad změny hesla:

```

//*****
import java.util.*;
import java.io.*;
import com.ibm.edms.od.*;

public class TcChangePassword
{
    public static void main ( String argv[] )
    {
        ODServer odServer;
        String server, userid, original_password, new_password;

        //-----
        // Je-li příliš málo parametrů, zobrazit syntaxi a odejít
        //-----
        if ( argv.length < 5 )
        {
            System.out.println( "použití: java TcChangePassword <server> <ID uživatele> <heslo> <nové heslo> <konfigurační adresář>
[-adresář lokálního serveru]" );
            return;
        }

        try
        {
            //-----
            // Vysvětlení
            //-----
            System.out.println( "Tento testovací případ by měl:" );
            System.out.println( " Přihlášení k serveru se zadaným heslem" );
            System.out.println( " Změna heslo na nové heslo" );
            System.out.println( " Odhlášení" );
            System.out.println( " Přihlášení k serveru s novým heslem" );
            System.out.println( " Změna hesla zpět na původní heslo" );
            System.out.println( " Odhlášení" );
            System.out.println( "" );
            System.out.println( "Jestliže testovací příkaz proběhne bez výjimky, není zapotřebí žádná další" );
            System.out.println( "analýza." );
            System.out.println( "" );
            System.out.println( "-----" );
            System.out.println( "" );

            //-----
            // Vytvoření zadaného serveru
            //-----
            server = argv[0];
            userid = argv[1];
            original_password = argv[2];
            new_password = argv[3];
            odServer = new ODServer( );
            odServer.initialize( argv[4], "TcChangePassword.java" );

            //-----
            // Přihlášení k serveru s původním heslem
            //-----
            System.out.println( "Přihlášení k " + server + " s původním heslem..." );
            if ( argv.length == 5 )
                odServer.logon( server, userid, original_password );
            else
                if ( argv.length == 6 )

```

```

        odServer.logon( server, userid, original_password, ODConstant.CONNECT_TYPE_LOCAL, 0, argv[5] );

//-----
// Změna na nové heslo a odhlášení
//-----
System.out.println( "Změna na nové heslo..." );
odServer.changePassword( new_password );
System.out.println( "Odhlášení..." );
odServer.logoff( );

//-----
// Přihlášení k serveru s novým heslem
//-----
System.out.println( "Přihlášení k " + serveru + " s novým heslem..." );
if ( argv.length == 5 )
    odServer.logon( server, userid, new_password );
else
    if ( argv.length == 6 )
        odServer.logon( server, userid, new_password, ODConstant.CONNECT_TYPE_LOCAL, 0, argv[5] );

//-----
// Změna zpět na původní heslo a odhlášení
//-----
System.out.println( "Změna na původní heslo..." );
odServer.changePassword( original_password );
System.out.println( "Odhlášení..." );
odServer.logoff( );

//-----
// Čištění
//-----
odServer.terminate( );
System.out.println( "" );
System.out.println( "-----" );
System.out.println( "" );
System.out.println( "Testovací případ dokončen úspěšně" );
System.out.println( "" );
}

catch ( ODEException e )
{
    System.out.println( "ODEException: " + e );
    System.out.println( "    id = " + e.getErrorId( ) );
    System.out.println( "    msg = " + e.getErrorMsg( ) );
    e.printStackTrace( );
}

catch ( Exception e2 )
{
    System.out.println( "výjimka: " + e2 );
    e2.printStackTrace( );
}
}
}

```

Dodatek E. Konverze AFP na HTML

Proces konverze AFP na HTML konvertuje dokumenty a prostředky AFP na dokumenty HTML. K procesu konverze AFP na HTML je zapotřebí služba AFP2WEB Transform od IBM Printing Systems Division. Službu AFP2WEB Transform musí na HTTP serveru instalovat a konfigurovat administrátor. Další informace o nabízené službě AFP2WEB Transform vám sdělí odpovědný pracovník IBM. Někdo ve vaší organizaci musí také zadat konfigurační volby pro dokumenty a prostředky AFP, které plánujete zpracovávat službou AFP2WEB Transform. Tato sekce popisuje, jak se zadávají konfigurační volby.

Poznámka: V tomto dokumentu jméno AFP2HTML.INI označuje konfigurační soubor. Chcete-li zadat soubor, který obsahuje konfigurační volby, viz téma “CONFIGFILE” na stránce 20.

Soubor AFP2HTML.INI uvádí konfigurační volby pro službu AFP2WEB Transform. Obvykle konfigurujete soubor AFP2HTML.INI s volbami pro určité aplikace AFP. Můžete však také uvést předvolené volby. Služba AFP2WEB Transform používá předvolené volby, když se konvertují dokumenty a prostředky pro aplikace AFP, které nejsou označeny v souboru AFP2HTML.INI. Chcete-li se dozvědět více podrobností o volbách a procesu konverze, prostudujte si dokumentaci ke službě AFP2WEB Transform.

Následující témata obsahují další informace o souboru AFP2HTML.INI:

- Formát souboru AFP2HTML.INI.
- Volby pro službu AFP2WEB Transform.
- Prohlížení konvertovaných dokumentů.

Poznámka: Chcete-li konvertovat dokumenty appletem AFP2HTML, musíte také zadat parametr `AFPVIEWING=HTML` do sekce `DEFAULT BROWSER` (nebo jiné sekce prohlížeče) v souboru `ARSWWW.INI`. Podrobnosti viz téma “AFPVIEWING” na stránce 30. (Jestliže plánujete používat rozhraní Retrieve Document API, musíte zadat parametr `_afp=HTML`. Podrobnosti najdete v tématu “Funkce Retrieve Document (Načtení dokumentu)” na stránce 71.) Musíte také zadat adresář, který obsahuje programy AFP2WEB Transform (viz téma “CONFIGFILE” na stránce 20).

Formát souboru AFP2HTML.INI

Toto je příklad souboru AFP2HTML.INI:

```
[CREDIT-CREDIT]
UseApplet=FALSE
ScaleFactor=1.0
CreateGIF=TRUE
SuppressFonts=FALSE
FontMapFile=creditFontMap.cfg
ImageMapFile=creditImageMap.cfg

[default]
ScaleFactor=1.0
CreateGIF=TRUE
SuppressFonts=FALSE
FontMapFile=fontmap.cfg
ImageMapFile=imagemap.cfg
```

Struktura souboru je podobná souboru Windows INI a obsahuje jeden objekt stanza pro každou aplikaci AFP a jeden předvolený objekt stanza. Řádek nadpisu objektu stanza označuje skupinu aplikací a samotnou aplikaci. Například řádek nadpisu:

```
[CREDIT-CREDIT]
```

Označuje skupinu aplikací CREDIT a aplikaci CREDIT. Pomocí – (pomlčka) oddělte jména v řádku nadpisu. Jména se musí shodovat se jmény skupin aplikací a samotných aplikací definovanými na serveru OnDemand. Jestliže skupina aplikací obsahuje více než jednu aplikaci, pak vytvořte jeden objekt stanza pro každou aplikaci.

Volby v objektu stanza [default] používá služba AFP2WEB Transform ke zpracování dokumentů pro aplikace AFP, které nejsou uvedeny v souboru AFP2HTML.INI. Předvolby se také používají, jestliže objekt stanza nějaké aplikace AFP neobsahuje jednu z voleb.

Volba UseApplet je v aplikaci ODWEK směrodatná. Určuje, zda se k zobrazení výstupu ze služby AFP2WEB Transform bude používat applet AFP2HTML. Předvolená hodnota je TRUE. Pokud uvedete hodnotu FALSE (applet AFP2HTML se nepoužívá k zobrazení výstupu), pak výstup formátuje a zobrazuje webový prohlížeč.

Pět zbývajících voleb je pro službu AFP2WEB Transform směrodatných. “Volby pro službu AFP2WEB” krátce popisuje, jak je služba AFP2WEB Transform používá.

Volby pro službu AFP2WEB

Tabulka 14 uvádí volby, které můžete zadat do souboru AFP2HTML.INI pro konverzi dokumentů pomocí služby AFP2WEB Transform.

Tabulka 14. Volby pro službu AFP2WEB Transform

Volby v souboru AFP2HTML.INI	Popis
AllObjects	Určuje, jak aplikace ODWEK zpracovává dokumenty, které jsou uloženy jako velké objekty na serveru OnDemand. Předvolená hodnota je 0 (nula) a znamená, že aplikace ODWEK načte pouze první segment dokumentu. Jestliže uvedete 1 (jedna), pak aplikace ODWEK načte všechny segmenty a před odesláním dokumentu klientovi je zkonvertuje. Poznámka: Jestliže povolíte podporu velkých objektů pro velmi velké dokumenty, pak vaši uživatelé mohou pocítit výraznou prodlevu, než si budou moci na klientovi prohlédnout dokument.
ScaleFactor	Upravuje velikost výstupu podle zadaného činitele zvětšení. Předvolená hodnota je 1.0. Když například zadáte hodnotu ScaleFactor=2.0, výstup je dvakrát větší než předvolená velikost; zadáním hodnoty ScaleFactor=0.5 je výstup poloviční než předvolená velikost. Předvolená velikost je odvozena od nastavení Zoom na stránce Logical Views v aplikaci OnDemand.
SuppressFonts	Určuje, zda se textové řetězce AFP převádějí. Jestliže zadáte SuppressFonts=TRUE a libovolný text, který používá font uvedený v souboru mapy fontů, se nepřevádí. Předvolená hodnota je FALSE, což znamená, že všechny textové řetězce AFP se převádějí. Soubor tabulky fontů je označen volbou FontMapFile
FontMapFile	Označuje jméno úplné cesty k souboru mapy fontů. Soubor mapy fontů obsahuje seznam fontů, které vyžadují zvláštní zpracování. Předvolený soubor mapy fontů má jméno imagfont.cfg a je umístěn v adresáři, který obsahuje programy AFP2WEB Transform. Podrobnosti o souboru mapy fontů najdete v dokumentaci ke službě AFP2WEB Transform.

Tabulka 14. Volby pro službu AFP2WEB Transform (pokračování)

Volby v souboru AFP2HTML.INI	Popis
ImageMapFile	Označuje soubor pro mapování fontů. Soubor pro mapování obrázků lze použít k odstranění obrázků z výstupu, k vylepšení vzhledu stínovaných obrázků a k nahrazení existujících obrázků obrázky vytvořenými službou AFP2WEB Transform. Mapování obrázků, které jsou ve vašich dokumentech AFP běžné (například logo firmy) snižuje čas potřebný pro přenos dokumentů. Jestliže je zadáno, soubor mapování obrázků musí existovat v adresáři, který obsahuje programy AFP2WEB Transform. Podrobnosti o souboru pro mapování obrázků najdete v dokumentaci ke službě AFP2WEB Transform.

Poznámka: Při konverzi dokumentů aplikace ODWEK odesílá následující volby službě AFP2WEB Transform. Tyto volby nejsou uváděny v souboru AFP2HTML.INI.

- Orientace. Určuje hodnotu rotace, která se má použít při prohlížení dokumentu. Předvolená hodnota je odvozena od nastavení Orientation na straně View Information v aplikaci OnDemand.
- Barva obrázku. Určuje barvu, která se má použít při prohlížení obrázků a grafiky. Předvolená hodnota je odvozena od nastavení Image Color na straně Logical Views v aplikaci OnDemand.

Prohlížení konvertovaných dokumentů

Volba UseApplet v souboru AFP2HTML.INI je pro aplikaci ODWEK směrodatná a určuje, zda se k zobrazení konvertovaného výstupu použije applet AFP2HTML. Předvolená hodnota je TRUE. Pokud uvedete hodnotu FALSE (applet AFP2HTML se nepoužívá k zobrazení výstupu), pak výstup formátuje a zobrazuje webový prohlížeč.

Obecně vzato, IBM doporučuje, abyste k zobrazení konvertovaného dokumentu vždy používali applet AFP2HTML. Jestliže dokument byl uložen do serveru OnDemand jako velký objekt, pak applet AFP2HTML přidá řídicí prvky, aby uživatelům pomohl snadno přecházet na libovolnou stranu dokumentu.

Dodatek F. Konverze AFP na PDF

Služba AFP2PDF Transform konvertuje dokumenty a prostředky AFP na dokumenty PDF. AFP2PDF Transform se služba nabízená od IBM Printing Systems Division. Administrátor musí instalovat a konfigurovat AFP2PDF Transform na HTTP serveru. Další informace o nabízené službě AFP2PDF Transform vám sdělí odpovědný pracovník IBM. Někdo ve vaší organizaci musí také uvést konfigurační volby pro dokumenty a prostředky AFP, které plánujete zpracovat s aplikací AFP2PDF Transform. Tato sekce popisuje, jak se zadávají konfigurační volby.

Poznámka: V tomto dokumentu jméno AFP2PDF.INI označuje konfigurační soubor. Chcete-li zadat soubor, který obsahuje konfigurační volby, viz téma “CONFIGFILE” na stránce 22.

Soubor AFP2PDF.INI obsahuje konfigurační volby pro službu AFP2PDF Transform. Obvykle konfigurujete soubor AFP2PDF.INI s volbami pro určité aplikace AFP. Můžete však také uvést předvolené volby. Služba AFP2PDF Transform používá předvolené volby, když se konvertují dokumenty a prostředky pro aplikace AFP, které nejsou označeny v souboru AFP2PDF.INI. Chcete-li se dozvědět více podrobností o volbách a procesu konverze, hledejte v dokumentaci ke službě AFP2PDF Transform.

Následující témata obsahují další informace o souboru AFP2PDF.INI:

- Zadání souboru AFP2PDF.INI
- Prohlížení konvertovaných dokumentů

Poznámka: Chcete-li konvertovat dokumenty, musíte také zadat parametr `AFPVIEWING=PDF` do sekce `DEFAULT BROWSER` (nebo jiné sekce prohlížeče) v souboru `ARSWWW.INI`. Podrobnosti viz téma “AFPVIEWING” na stránce 30. (Jestliže plánujete používat rozhraní Retrieve Document API, musíte zadat parametr `_afp=PDF`. Podrobnosti najdete v tématu “Funkce Retrieve Document (Načtení dokumentu)” na stránce 71.)

Zadání souboru AFP2PDF.INI

Níže je uveden příklad souboru AFP2PDF.INI:

```
[CREDIT-CREDIT]
OptionsFile=
ImageMapFile=creditImageMap.cfg
```

```
[default]
OptionsFile=
ImageMapFile=imagemap.cfg
AllObjects=0
```

Struktura souboru je podobná souboru Windows INI a obsahuje jeden objekt stanza pro každou aplikaci AFP a jeden předvolený objekt stanza. Řádek nadpisu objektu stanza označuje skupinu aplikací a samotnou aplikaci. Například řádek nadpisu:

```
[CREDIT-CREDIT]
```

Označuje skupinu aplikací CREDIT a aplikaci CREDIT. Pomocí – (pomlčka) oddělte jména v řádku nadpisu. Jména se musí shodovat se jmény skupin aplikací a samotných aplikací definovaných na serveru OnDemand. Jestliže skupina aplikací obsahuje více než jednu aplikaci, pak vytvořte jeden objekt stanza pro každou aplikaci.

Parametry, které uvedete v objektu stanza [default], se používají službou AFP2PDF Transform ke zpracování dokumentů pro aplikace AFP, které neuvádí soubor AFP2PDF.INI. Předvolené parametry se také používají, jestliže objekt stanza v aplikaci AFP neobsahuje jeden ze zadaných parametrů.

Parametr OptionsFile uvádí jméno úplné cesty k souboru, jež obsahuje volby pro převod používané službou AFP2PDF Transform. Volby pro převod se používají pro dokumenty AFP, které vyžadují speciální zpracování. Podrobnosti o souboru voleb pro převod najdete v dokumentaci ke službě AFP2PDF Transform.

Parametr ImageMapFile označuje soubor pro mapování obrázků. Soubor pro mapování obrázků lze použít k odstranění obrázků z výstupu, k vylepšení vzhledu stínovaných obrázků a k nahrazení existujících obrázků obrázky vytvořenými službou AFP2PDF Transform. Mapování obrázků, které jsou společné ve většině vašich dokumentů AFP (například logo firmy) snižuje čas potřebný pro přenos dokumentů. Jestliže je zadáno, soubor mapování obrázků musí existovat v adresáři, který obsahuje programy AFP2PDF Transform. Chcete-li zadat adresář, který obsahuje tyto programy pro službu AFP2PDF Transform, viz téma "INSTALLDIR" na stránce 22. Podrobnosti o souboru pro mapování obrázků najdete v dokumentaci ke službě AFP2PDF Transform.

Parametr AllObjects určuje, jak aplikace ODWEK zpracovává dokumenty, které jsou uloženy jako velké objekty na serveru OnDemand. Předvolená hodnota je 0 (nula) a znamená, že aplikace ODWEK načte pouze první segment dokumentu. Jestliže uvedete 1 (jedna), pak aplikace ODWEK načte všechny segmenty a před odesláním dokumentu klientovi je zkonvertuje. **Poznámka:** Jestliže povolíte podporu velkých objektů pro velmi velké dokumenty, pak vaši uživatelé mohou pocítit výraznou prodlevu, než si budou moci na klientovi prohlédnout dokument.

Prohlížení konvertovaných dokumentů

Chcete-li zobrazovat konvertované dokumenty prohlížečem Adobe Acrobat, musíte jej získat pro webové prohlížečské programy, které se používají ve vaší organizaci.

Dodatek G. Konfigurační soubory HTTP serveru

Tato část obsahuje příklady konfiguračních souborů následujících HTTP serverů:

- HTTP Original Server (již není podporován)
- HTTP Apache Server
- WebSphere Application Server

Poznámka: Konfiguraci vám usnadní dokumentace k HTTP.

HTTP Apache Server

Poznámky:

1. Na řádku `Listen` nahraďte řetězec **n.n.n.n** IP adresou počítače iSeries, na kterém je spuštěný software serveru OnDemand.
2. V řádku `Listen` nahraďte **p** číslem portu TCP/IP používaného pro komunikaci HTTP serveru a klienta. Předvolené číslo portu TCP/IP je 80.
3. V řádku `DefaultNetCCSID` nahraďte řetězec **xxxx** hodnotou CCSID z instance OnDemand.
4. Na řádku `ServerName` nahraďte řetězec **f.q.h.n** plně kvalifikovaným jménem TCP/IP hostitele iSeries, na kterém je spuštěný software serveru OnDemand.
5. V případě jazyků DBCS změňte řádek `CGIConvMode` na:

```
CGIConvMode %%EBCDIC_JCD/MIXED%%
```

```
#HTTP Apache Server Configuration for the ODWAPACHE server
```

```
Listen n.n.n.n:p
DefaultNetCCSID xxxx
HostNameLookups off
RuleCaseSense OFF
LimitRequestBody 102400
ErrorLog /www/odwapache/logs/error_log
ServerName f.q.h.n
DocumentRoot /www/odwapache
UseCanonicalName Off
CGIConvMode %%EBCDIC/MIXED%%
ScriptLogLength 200
ScriptLog /www/odwapache/logs/cgi_log
Alias /logon /www/odwapache/odw_logon.htm
AliasMatch ^/images/(.*)$ /QIBM/UserData/OnDemand/www/images/$1
AliasMatch ^/applets/com/ibm/edms/od/(.*)$ /QIBM/ProdData/OnDemand/www/applets/$1
AliasMatch ^/applets/(.*)$ /QIBM/ProdData/OnDemand/www/applets/$1
ScriptAliasMatch ^/scripts/arswww.cgi$ /QSYS.LIB/QRDARS.LIB/ARS3WCGI.PGM
ScriptAliasMatch ^/scripts/arswww\.cgi/(.*)$ /QSYS.LIB/QRDARS.LIB/ARS3WCGI.PGM
AddType www/unknown cab
AddType www/unknown jar
AlwaysDirectoryIndex On
DirectoryIndex index.html
<Directory/>
  Options None
  Options +ExecCGI
  AllowOverride None
  Order deny,allow
  Deny from all
  <LimitExcept GET HEAD OPTIONS TRACE POST PUT>
  </LimitExcept>
</Directory>
<Directory /QIBM/ProdData/OnDemand/www/applets/>
  Order allow,deny
```

```
    Allow from All
  </Directory>
<Directory /QIBM/ProdData/OnDemand/www/images/>
  Order allow,deny
  Allow from All
  Options +FollowSymLinks
</Directory>
<Directory /QIBM/ProdData/OnDemand/www/samples/>
  Order allow,deny
  Allow from All
  Options +FollowSymLinks
</Directory>
<Directory /www/odwapache/>
  Order allow,deny
  Allow from All
</Directory>
<Directory /QSYS.LIB/QRDARS.LIB>
  Order allow,deny
  Allow from All
  Options +ExecCGI
</Directory>
```

| **WebSphere Application Server**

| Vzorový konfigurační soubor WebSphere lze získat na webové stránce podpory produktu
| IBM Content Manager OnDemand for iSeries na adrese
| <http://www.ibm.com/software/data/ondemand/400/support.html>. Příklad *ODWEK WebSphere*
| najdete v kategorii Technotes.

Dodatek H. Žádný HTML výstup

Aplikace ODWEK používá direktivu `_nohtml` k určení typu výstupu generovaného nějakou funkcí (například Logon). Standardně se jedná o subsystém HTML. Jestliže uvedete specifi `_nohtml=1`, aplikace ODWEK pak generuje výstup ASCII s oddělovači. Tato kapitola popisuje výstup ASCII s oddělovači generovaný aplikací ODWEK.

Výstup ASCII s oddělovači

Výstup ASCII s oddělovači generovaný aplikací ODWEK je sada výstupních záznamů, které obsahují řetězcové hodnoty, klíčová slova a oddělovače a separátory funkcí, záznamů a řetězců:

- Znakové řetězcové hodnoty jsou výstupní data z funkce, jiná než klíčová slova, oddělovače a separátory. Znaková řetězcová hodnota je například další volaná funkce, jméno složky, jméno pole složky, vyhledávací operátor a hodnota pole.
- Klíčová pole se skládají z určitého znakového řetězce. Klíčová slova jsou například ACTION, DOC, FOLDER, NUMROWS a ROW.
- Oddělovače funkcí se skládají ze specifických znakových řetězců [BEGIN] a [END].
- Oddělovač záznamu je znak nového řádku `\n`. Všechny záznamy jsou odděleny znakem nového řádku.
- Jako oddělovač a separátor řetězců slouží standardně stříška (`^`) a levá závorka (`(`) a pravá závorka (`)`). Příklad:

```
[folderName^folderDesc]
```

Jestliže klíčové slovo obsahuje více než jednu znakovou řetězcovou hodnotu, pak jsou hodnoty odděleny stříškou. Každá klíčová sada znakových řetězcových hodnot je oddělena levou a pravou závorkou.

Některé hodnoty znakových řetězců mohou být uloženy v seznamu, odděleny stříškou a uzavřeny v pravé a levé závorce. Například seznam platných vyhledávacích operátorů pro pole může vypadat následovně:

```
[1^2^4^8^16^32]
```

Znaky předvolené jako oddělovače a separátory řetězců můžete potlačit. Podrobnosti viz téma “[NO HTML]” na stránce 28.

- Jednotlivá hodnota null znakového řetězce se zadává tím, že není uvedena žádná hodnota mezi dvěma uvozovkami (`""`). Prázdný seznam (null) se zadává tím, že chybí hodnota mezi levou a pravou závorkou (`()`).

Přihlášení

Níže je uveden příklad výstupu ASCII s oddělovači generovaný funkcí Logon:

```
[BEGIN]\nACTION=searchCriteriaUrl\nFOLDER=[folderName^folderDesc]\nFOLDER=[folderName^folderDesc]\n
```

```
⋮
```

```
[END]\n
```

Poznámky

1. Řetězec `searchCriteriaUrl` uvádí jméno další funkce, která má být spuštěna a její parametry.
2. Řetězec `folderName` uvádí jméno složky. Jméno není uvedené v uvozovkách.
3. Řetězec `folderDesc` je popis složky. Popis se neuvádí v uvozovkách.

Vyhledávací kritéria

Níže je uveden příklad dat ASCII s oddělovači generovaných funkcí Search Criteria:

```
[BEGIN]\nACTION=hitListUrl\nDISPLAY_ORDER=[field1^field2^...fieldN]\nNUMROWS=numberOfRows\nROW=[criteriaName^[[validOp]^defOp]^[inpType^inpAssocData]\n\n:\n[END]\n
```

Poznámky

1. Řetězec `hitListUrl` uvádí jméno další funkce, která má být spuštěna a její parametry.
2. Klíčové slovo `DISPLAY_ORDER` uvádí pořadí, v jakém se pole složky mají zobrazovat.
3. Řetězec `numberOfRows` uvádí počet záznamů klíčových slov `ROW`, které následují. Funkce generuje jeden záznam klíčového slova `ROW` pro každé vyhledávací pole.
4. Řetězec `criteriaName` představuje vyhledávací kritéria pro vyhledávací pole. Vyhledávací kritéria se neuvádějí v uvozovkách.
5. Řetězec `validOp` je seznam celočíselných hodnot, které představují platné vyhledávací operátory pro vyhledávací pole:
 - 1 Equal (Rovno)
 - 2 Not Equal (Nerovno)
 - 4 Less Than (Menší než)
 - 8 Less Than or Equal (Menší než nebo rovno)
 - 16 Greater Than (Větší než)
 - 32 Greater Than or Equal (Větší než nebo rovno)
 - 64 In (V)
 - 128 Not In (Ne v)
 - 256 Like (Podobné)
 - 512 Not Like (Nepodobné)
 - 1024 Between (Mezi)
 - 2048 Not Between (Ne mezi)
6. Řetězec `defOp` je celočíselná hodnota představující předvolený vyhledávací operátor.
7. Řetězec `inpType` představuje typ vyhledávacího pole:
 - A Prohledávání textu anotací
 - C Výběr
 - N Normální
 - S Segment
 - T Textové prohledávání
 - Z Vyhledávání barvy anotace
8. Řetězec `inpAssocData` je seznam přidružený k `defOp` a `inpType`. Viz téma Tabulka 15 na stránce 133.

Tabulka 15. Předvolený operátor a typ vstupu přidružený k parametru *inpAssocData*

defOp	inpType	inpAssocData
Between, Not Between (Mezi, ne mezi)	N	Null: [] nebo seznam: [defaultField1 ^... ^defaultFieldN] Příklad: ["01/31/96" ^ "01/31/97"] ["01/31/96" ^ ""] ["" ^ "01/31/97"]
Jiné platné operátory	A, N, T, Z	Null: [] nebo jednotlivá řetězcová hodnota, která představuje předvolenou hodnotu pole
Jiné platné operátory	C, S	[[listOfChoices] ^ defaultChoice] Příklad: [["JFIF" ^ "TIFF" ^ "PCX"] ^ "TIFF"] [["JFIF" ^ "TIFF" ^ "PCX"] ^ ""]

Seznam nalezených dokumentů

Níže je uveden příklad výstupu ASCII s oddělovači generovaný funkcí Document Hit List:

```
[BEGIN]\n
ACTION=hitListURL\n
MSG=Only 20 documents can be listed for this folder.
DOC=[criteria1^criteria2^criteriaN^docid^fileType^docLocation]\n
:
[END]\n
```

Poznámky

1. Řetězec *hitListURL* uvádí jméno další funkce, která se bude spouštět, a parametry pro funkci.
2. Klíčové slovo *MSG* uvádí příklad chybové zprávy ve výstupu ASCII s oddělovači. Aplikace ODWEK standardně zasílá klientovi chybové zprávy. Když však funkce obsahuje direktivu *_nohtml=1*, aplikace ODWEK místo toho generuje text zprávy ve výstupu ASCII s oddělovači.
3. Řetězce *criteria1*, *criteria2*, a *criteriaN* představují hodnoty vyhledávacích kritérií. Hodnoty jsou uvedeny v pořadí, v jakém se objevují v seznamu dokumentů. Hodnoty nejsou uváděny v uvozovkách.
4. Řetězec *docid* je identifikátor dokumentu.
5. Řetězec *fileType* označuje typ dat dokumentu:
 - A AFP
 - B BMP
 - E Email
 - F JFIF
 - G GIF
 - L Line
 - N None
 - O OD Defined
 - P PDF
 - T TIFF
 - U uživatelsky definovaný
 - X PCX
6. Řetězec *docLocation* označuje úložiště dokumentu:

0	Neznámé
1	Rychlá vyrovnávací paměť OnDemand
2	Archivní paměť
3	Externí rychlá vyrovnávací paměť

Zobrazení anotací

Níže je uveden příklad výstupu ASCII s oddělovači generovaný funkcí View Annotations:

```
[BEGIN]\nNOTE 4: 15:42:44 PM Mountain Standard Time Thursday November 19, 1998...\nPublic - Cannot be copied to another server\nTest note from the OnDemand Internet Client.\n[END]\n
```

Chybová zpráva

Níže je uveden příklad výstupu ASCII s oddělovači generovaného, když se vyskytnou chyby:

```
[ERROR]\nID=nnnn\nMSG=errorMessageText\n
```

Poznámky

1. Řetězec nnnn je číslo chybové zprávy.
2. Řetězec errorMessageText je text chybové zprávy.

Dodatek I. Podpora národního jazyka

Tato část obsahuje informace, které pomáhají administrátorům konfigurovat ODWEK pro jazyky DBCS.

Parametry CODEPAGE a LANGUAGE v souboru ARSWWW.INI se používají k zadání konfiguračních voleb National Language (NL).

Parametr CODEPAGE označuje kódovou stránku serveru ODWEK a je nutné, aby byl kompatibilní s databází OnDemand na serveru knihoven OnDemand. Parametr CODEPAGE je zapotřebí zadat **pouze** tehdy, jestliže se kódová stránka pracovní stanice, na které spouštíte aplikaci ODWEK, liší od kódové stránky databáze OnDemand na serveru knihoven OnDemand. Systém používá kódovou stránku pracovní stanice, na které se spouští aplikace ODWEK, jako předvolenou hodnotu.

Parametr LANGUAGE určuje katalog zpráv, který ODWEK používá k zobrazení zpráv.

Tabulka 16 uvádí seznam kódových stránek a jazyků, které jsou podporovány na OnDemand. Sloupec **CODEPAGE=** uvádí hodnotu pro kódovou stránku a je nutné jej zadat **pouze** tehdy, jestliže kódová stránka pracovní stanice, na které spouštíte svou aplikaci ODWEK, se liší od kódové stránky databáze OnDemand. Sloupec **LANGUAGE=** uvádí hodnoty, které jsou přiřazeny k přeloženému katalogu zprávy.

Poznámka: Linux není uveden v tabulce, protože Linux nemůže být cílový server pro aplikaci ODWEK. To znamená, že ačkoli ODWEK může být spuštěn v systému Linux, databáze OnDemand (server knihoven OnDemand) v systému Linux spuštěn být nemůže.

Tabulka 16. kódové stránky DBCS, jazyky, znakové sady a lokální nastavení

Území	LANGUAGE=	OS	Kódová stránka databáze	CODEPAGE=	Kódová sada	Lokalita
Čína (PRC)	CHS	AIX	1383	1383	IBM_eucCN	zh_CN
		HP-UX	1383	1383	hp15CN	zh_CN. hp15CN
		Solaris	1383	1383	gb2312	zh
		Windows	1386	1386	GBK	—
		z/OS nebo OS/390 (EBCDIC)	935	935	IBM-935	—
Japonsko	JPN	AIX	954	954	IBM_eucJP	ja_JP
		HP-UX	954	954	eucJP	ja_JP.eucJP
		Solaris	954	954	eucJP	ja
		Windows	943	943	IBM-943	—
		z/OS nebo OS/390 (EBCDIC)	939	939	IBM-939	—

Tabulka 16. kódové stránky DBCS, jazyky, znakové sady a lokální nastavení (pokračování)

Území	LANGUAGE=	OS	Kódová stránka databáze	CODEPAGE=	Kódová sada	Lokalita
Jižní Korea	KOR	AIX	970	970	IBM_eucKR	ko_KR
		HP-UX	970	970	eucKR	ko_KR.eucKR
		Solaris	970	970	5601	ko
		Windows	1363	1363	1363	—
		z/OS nebo OS/390 (EBCDIC)	933	933	IBM-933	—
Taiwan	CHT	AIX	964	964	IBM_eucTW	zh_TW
		HP-UX	964	964	eucTW	zh_TW.eucTW
		Solaris	964	964	cns11643	zh_TW
		Windows	950	950	big5	—
		z/OS nebo OS/390 (EBCDIC)	937	937	IBM-937	—

Další informace o konfiguraci systému OnDemand pro jazyky DBCS najdete v tématu “Podpora národních jazyků” v publikaci *Plánování a instalace*.

Dodatek J. Nástroje k určování problémů

Ke shromáždění informací o systému a dokumentech můžete použít nástroje uvedené v Tabulka 17. Tyto informace můžete použít pro řešení problémů, které máte s konfigurací aplikace ODWEK, a při pomoci jiným pracovníkům ve vaší organizaci, kteří mají problémy s používáním appletů a programů typu plugin.

Tabulka 17. Nástroje pro zjišťování problémů

Nástroj	Účel	Jak zapnout
Výstup HTML	Uložte kopii souboru HTML, který aplikace ODWEK vrací do prohlížeče.	Z menu Soubor v prohlížeči vyberte položku Uložit jako.
Soubory protokolu na serveru	Ukládají informace o přístupu, chybách a serveru.	<p>Proveďte následující:</p> <ol style="list-style-type: none"> V sekci DEBUG v souboru ARSWWW.INI nastavte parametr LOG na hodnotu 1 (jedna). Soubor protokolu, který generuje aplikace ODWEK, má jméno ARSWWW.LOG a zapisuje se do adresáře uvedeného parametrem LOGDIR. (Předvolený adresář je /QIBM/UserData/OnDemand/WWW/LOG.) Důležité: Jestliže je uvedena sekce DEBUG, musí to být první spustitelný příkaz v souboru ARSWWW.INI. Konfigurujte protokolování na svém HTTP serveru. (Každý HTTP server může mít jiný způsob, jak se konfiguruje protokolování, a může mít rozdílné protokoly a volby, kterými můžete povolit shromažďování více nebo méně podrobných informací.) <p>Poznámka: Do souboru protokolu může být zapsáno významné množství informací, a proto IBM doporučuje, abyste protokolování povolovali pouze tehdy, když je to potřebné, například když zkoušíte znovu vyvolat problém. Pokud potřebujete povolit protokolování na delší dobu, ověřte, zda cesty protokolovacích souborů vedou na paměťová zařízení s dostatkem volného prostoru. Také pravidelně odstraňujte staré soubory protokolu ze systému.</p>
Java Console	Zobrazuje zprávy, které generují applety.	<ul style="list-style-type: none"> Netscape: V menu Netscape Communicator vyberte volbu Tools a volbu Java Console. Internet Explorer: <ol style="list-style-type: none"> V menu Nástroje vyberte volbu Možnosti internetu. Na straně Upřesnit vyberte volbu Konzole jazyka Java. Restartujte prohlížeč. V menu Upřesnit vyberte volbu Konzole jazyka Java.

Tabulka 17. Nástroje pro zjišťování problémů (pokračování)

Nástroj	Účel	Jak zapnout
AFP Web Viewer Trace Facility	Zachycuje podrobné informace o dokumentech AFP prohlížených webovým prohlížečem AFP.	<p>Ověřte, zda následující sekce existuje v souboru FLDPORT2.INI na pracovní stanici uživatele:</p> <pre>[Misc] ViewTraceFile=d:\temp\afpplgin.log Trace=TRUE</pre> <p>Ověřte cestu souboru protokolu. Pamatujte na to, abyste vypnuli protokolování, až shromáždíte informace, které potřebujete.</p>
Protokolování systému OnDemand	Ukládá systémové zprávy (například přihlašování a odhlašování) a zprávy ze skupiny aplikací, které souvisejí s dokumenty (například dotaz a načtení) a anotace.	<p>Proveďte následující:</p> <ol style="list-style-type: none"> 1. Povolte protokolování systému a skupiny aplikací pro server OnDemand. Pomocí administrativního klienta aktualizujte systémové parametry pro server. 2. Povolte specifické zprávy skupiny aplikací, kterou chcete protokolovat. Pomocí administrativního klienta aktualizujte volby protokolování zpráv pro skupinu aplikací.

Poznámky

Tyto informace se týkají produktů a služeb nabízených v USA.

IBM nemusí nabízet produkty, služby a funkce popsané v tomto dokumentu v jiných zemích. Informace o produktech a službách, které jsou v současné době k dispozici ve vaší oblasti, získáte od místního zástupce IBM. Žádný odkaz na produkt, program nebo službu IBM neznámá ani z něj nelze vyvozovat, že smí být použit pouze tento produkt, program nebo služba IBM. Použit lze jakýkoliv funkčně ekvivalentní produkt, program nebo službu neporušující práva IBM na duševní vlastnictví. Za vyhodnocení a ověření činnosti libovolného produktu, programu či služby jiného výrobce než IBM však odpovídá uživatel.

IBM může mít patenty nebo podané žádosti o patent, které zahrnují předmět tohoto dokumentu. Získání tohoto dokumentu uživateli neposkytuje licenci na tyto patenty. Písemné dotazy na licence můžete posílat na adresu:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

Pokud máte zájem o licenci v zemi s dvoubajtovou znakovou sadou (DBCS), kontaktujte zastoupení IBM ve Vaší zemi nebo pošlete písemný dotaz na adresu:

IBM World Trade Asia Corporation
Licensing
2-31 Roppongi 3-chome, Minato-ku
Tokyo 106, Japan

Následující odstavec se netýká Velké Británie nebo kterékoliv jiné země, kde taková opatření odporují místním zákonům:

SPOLEČNOST INTERNATIONAL BUSINESS MACHINES POSKYTUJE TUTO PUBLIKACI TAKOVOU, JAKÁ JE ("AS-IS"), BEZ JAKÝCHKOLIV ZÁRUK, VYJÁDŘENÝCH NEBO ODVOZENÝCH, VČETNĚ, MIMO JINÉ, ODVOZENÝCH ZÁRUK NEPORUŠENÍ PRÁV TŘETÍCH STRAN, ZÁRUKY PRODEJNOSTI NEBO VHODNOSTI PRO URČITÝ ÚČEL.

Některé státy nepovolují odmítnutí vyjádřených nebo odvozených záruk při určitých transakcích, a proto se vás předchozí prohlášení nemusí týkat.

Tato publikace může obsahovat technické nepřesnosti nebo typografické chyby. Zde uvedené informace jsou pravidelně aktualizovány; změny budou začleněny do nových vydání publikace. Společnost IBM má právo kdykoliv bez upozornění zdokonalovat nebo měnit produkty a programy popsané v této publikaci.

Jakékoliv odkazy v této publikaci na webové stránky jiných společností než IBM jsou poskytovány pouze pro větší pohodlí uživatele a nemohou být žádným způsobem vykládány jako doporučení těchto webových stránek společností IBM. Materiály na těchto webových stránkách nejsou součástí materiálů pro daný produkt IBM a používání těchto webových stránek je na vaše vlastní nebezpečí.

IBM může použít nebo distribuovat jakékoliv informace, které jí sdělíte, libovolným způsobem, který považuje za odpovídající, aniž by jí tím vznikl jakýkoliv závazek vůči Vám.

Ten, kdo si přeje získat informace o licencích k tomuto programu za účelem umožnit: (i) výměnu informací mezi nezávisle vytvořenými programy a ostatními programy (včetně tohoto) a (ii) společné použití vyměněných informací, by měl kontaktovat koordinátora softwarové interoperability na adrese:

IBM Corporation
Software Interoperability Coordinator
3605 Highway 52 N
Rochester, MN 55901–7829
U.S.A.

Informace tohoto typu mohou být za odpovídajících podmínek dostupné. V některých případech připadá v úvahu zaplacení poplatku.

Zde popsaný licencovaný program a všechny licencované materiály, které jsou pro něj k dispozici, poskytuje IBM podle podmínek smlouvy IBM Customer Agreement nebo jiné ekvivalentní dohody mezi IBM a zákazníkem.

Informace týkající se produktů jiných firem než IBM byly získány od dodavatelů těchto produktů, z jejich publikovaných sdělení, nebo z jiných veřejně dostupných zdrojů. IBM netestovala tyto produkty a nemůže tudíž potvrdit spolehlivost, kompatibilitu a další prohlášení vztahující se k produktům od jiných dodavatelů. Dotazy, které se týkají vlastností produktů jiných firem než IBM, musí být adresovány jejich dodavatelům.

LICENČNÍ INFORMACE:

Tyto informace obsahují vzorové aplikační programy, uvedené ve zdrojovém jazyce, které demonstrují techniku programování na různých operačních systémech. Tyto vzorové programy můžete kopírovat, modifikovat a distribuovat v jakékoliv formě za účelem vývoje, používání, prodeje nebo distribuce aplikačních programů, podřizujících se aplikačnímu programovému rozhraní pro daný operační systém, pro který byly tyto vzorové programy napsány, aniž byste museli platit IBM poplatky. Tyto vzorové programy nebyly důkladně testovány za všech podmínek. IBM proto nemůže zaručit nebo potvrdit spolehlivost, obsluhovatelnost nebo funkčnost těchto produktů. Tyto vzorové programy můžete kopírovat, měnit a distribuovat jakýmkoliv způsobem, aniž by Vám vznikl finanční závazek vůči IBM, a to pro účely vývoje, užívání, marketingu nebo distribuce aplikačních programů, které jsou v souladu s rozhraními API od IBM.

Pokud si tuto publikaci prohlížíte ve formě softcopy, nemusí se zobrazit fotografie a barevné ilustrace.

Ochranné známky a servisní značky

Advanced Function Presentation, AFP, AS/400, IBM, iSeries, Operating System/400, OS/390, OS/400, Redbooks, WebSphere a z/OS are jsou ochranné známky společnosti International Business Machines Corporation ve Spojených státech anebo jiných zemích.

Java a všechny ochranné známky a loga obsahující slovo Java jsou ochrannými známkami nebo registrovanými ochrannými známkami společnosti Sun Microsystems, Inc. ve Spojených státech anebo jiných zemích.

Lotus je ochranná známka společnosti Lotus Development Corporation ve Spojených státech anebo jiných zemích.

Microsoft, Windows a Windows NT jsou registrované ochranné známky společnosti Microsoft Corporation ve Spojených státech anebo jiných zemích.

Další jména společností, produktů nebo služeb mohou být ochrannými známkami nebo servisními značkami jiných společností.

Rejstřík

A

- adresář obrázků 16
- adresář paměti cache 14
- AFP Web Viewer
 - fonty 46
 - fonty AFP 46
 - instalace 41
 - instalace uživatelsky definovaných souborů 43
 - konfigurace 47
 - mapování fontů AFP 46
 - o aplikaci 1
 - požadavky 42
 - přizpůsobení instalace 43
 - uživatelsky definované soubory 43
- AFP2PDF Transform
 - instalace 9
 - konfigurace 127
 - povolení 21
- AFP2WEB Transform
 - instalace 9
 - konfigurace 123
 - povolení 20
- aktualizace dokumentu
 - API 76
 - Java API 117
 - parametry 76
 - popis funkce 6
 - vzorové volání funkce 77
- anotace 29, 35
 - API 54, 78
 - Java API 113, 115
 - parametry 54, 78
 - popis funkce 5, 6
 - výstup ASCII s oddělovači 134
 - vzorové volání funkce 56, 79
- API
 - aktualizace dokumentu 76
 - anotace 54, 78
 - diagnostické informace 88
 - načtení dokumentu 71
 - odhlášení 63
 - průvodce programováním 85
 - průvodce programováním rozhraní Java API 85
 - přidání anotace 54
 - přihlášení 65
 - příklady 85
 - reference 53, 81, 83
 - reference CGI API 53
 - reference pro rozhraní Java API 83
 - seznam nalezených dokumentů 59
 - tisk dokumentu 67
 - tisk na serveru 67
 - trasování a diagnostické informace 88
 - vyhledávací kritéria 74
 - vzorový kód 85
 - zacházení s výjimkami 88
 - změna hesla 57
 - zobrazení anotací 78
- aplikační programové rozhraní (API)
 - Viz API
- applety 13

- applety (*pokračování*)
 - adresář 22
 - instalace 41
 - o aplikaci 1
 - parametr APPLETACHEDIR 13
 - podpora velkých objektů 20, 125
 - povolení 22
 - požadavky 42
- atribut metody v návěští form 7

C

- CGI API
 - reference 53
- cookie 31, 42
- CREDIT.HTM 39

Č

- časový limit 19
- časový limit nečinnosti 19
- číslo portu 11, 12

D

- DBCS 135
- diagnostické informace 88
- dočasné úložiště 18
- dočasný pracovní adresář 18
- dokumenty
 - AFP 30
 - aktualizace s Java API 117
 - EMAIL 31
 - konverze 30, 31, 32
 - Line Data 32
 - načtení 30
 - odkazy 33
 - tisk s Java API 112
 - typ média 22
 - typ obsahu MIME 22
 - uložení do paměti cache 14
 - zobrazení 30, 31, 32
- dokumenty AFP
 - konverze 30
 - typ média 24
 - typ obsahu MIME 24
 - zobrazení 30
- dokumenty BMP
 - typ média 24
 - typ obsahu MIME 24
- dokumenty EMAIL
 - konverze 31
 - typ média 25
 - typ obsahu MIME 25
 - zobrazení 31
- dokumenty GIF
 - typ média 24
 - typ obsahu MIME 24
- dokumenty JFIF
 - typ média 25

- dokumenty JFIF (*pokračování*)
 - typ obsahu MIME 25
- dokumenty Line Data
 - konverze 32
 - typ média 25
 - typ obsahu MIME 25
 - zobrazení 32
- dokumenty PCX
 - typ média 26
 - typ obsahu MIME 26
- dokumenty PDF
 - typ média 26
 - typ obsahu MIME 26
- dokumenty TIFF
 - typ média 26
 - typ obsahu MIME 26
- dokumenty, Java API 107, 109

F

- fonty
 - AFP 46
 - mapování 46
 - TrueType 46
- fonty AFP
 - mapování 46
- fonty TrueType
 - mapování fontů AFP 46
- funkce
 - aktualizace dokumentu 6
 - anotace 5, 6
 - načtení dokumentu 6
 - odhlášení 6
 - přidání anotace 5
 - přihlášení 6
 - seznam nalezených dokumentů 5
 - tisk dokumentů 6
 - tisk dokumentů na serveru 6
 - vyhledávací kritéria 6
 - změna hesla 5
 - zobrazení anotací 6

H

- hesla
 - Java API 90, 120
- hierarchie balení, Java 85
- hledací hodnoty, Java API 94, 98
- hledání složky 94, 98, 100, 102, 107
- HTTP server
 - konfigurační soubory 129
 - soubor httpd.conf 129
- HTTP server Apache
 - konfigurační soubory 129
- HTTP server, software
 - instalace 9

CH

- chybová zpráva
 - výstup ASCII s oddělovači 134
- chyby 88, 137

I

- ID uživatele
 - Java API 90
 - uložení do paměti cache 15
- Image Web Viewer
 - instalace 41
 - konfigurace 47
 - o aplikaci 1
 - požadavky 42
- instalace
 - AFP Web Viewer 41
 - applety 41
 - HTTP server 9
 - Image Web Viewer 41
 - java applet AFP2HTML 41
 - java applet Line Data 41
 - java applety 41
 - OS/400 10
 - požadavky 9
 - programy typu plugin 41
 - přizpůsobení 43
 - software ODWEK 9
 - soubor ARSWWW.INI 10
 - uživatelská pracovní stanice 41
 - uživatelsky definované soubory 43

J

- Java API
 - o aplikaci 1
 - průvodce programováním 85
 - reference 83
 - softwarové požadavky 1
- java applet AFP2HTML
 - instalace 41
 - o aplikaci 1, 4
 - parametr APPLETCACHEDIR 13
 - podpora velkých objektů 20, 125
 - požadavky 9, 42
- java applet AFP2PDF
 - adresář 22
 - povolení 22
- java applet Line Data
 - instalace 41
 - o aplikaci 1, 4
 - parametr APPLETCACHEDIR 13
 - požadavky 42
- java applety
 - adresář 22
 - instalace 41
 - o aplikaci 1, 4
 - parametr APPLETCACHEDIR 13
 - podpora velkých objektů 20, 125
 - povolení 22
 - požadavky 42
- java prohlížeč Line Data Viewer
 - konfigurace 48
 - parametr ODApplet.jre.path.IE 33
 - parametr ODApplet.jre.path.NN 34
 - parametr ODApplet.jre.version 34
 - parametr ODApplet.version 34
- Java servlet
 - reference 81
- Java Virtual Machine 42
- jazyk 17, 135
- jazyky s dvoubajtovou znakovou sadou 135

jméno aplikace
 Java API 90
jméno hostitele 12
jméno složky, Java API 105
JVM 42

K

kódová stránka 16, 135
komunikační protokol TCP/IP 11, 12
komunikační protokoly 11, 12
konfigurace
 HTTP server 9
 konfigurační soubor AFP2HTML 123
 konfigurační soubor AFP2PDF 127
 software ODWEK 9
 soubor ARSWWW.INI 10
konfigurační soubor AFP2HTML 123
konfigurační soubor AFP2PDF 127

L

LOGON.HTM 39
lokální adresář
 Java API 90

M

mapování fontů AFP 46
maximální výskytů 33
metoda GET 7
metoda POST 7

N

načtení
 dokumenty 30
načtení dokumentu 109
 API 71
 parametry 71
 popis funkce 6
 vzorové volání funkce 73
nápověď 137
nastavení hesel 120
NLS 16, 17, 135

O

o aplikaci OnDemand Internet Connection 1
o této publikaci 1
ODCallback 111
ODCriteria
 aktualizace dokumentu 117
 dokumenty, aktualizace 117
 hledací hodnoty 94, 100, 102
 jméno 94
 operandy 94, 100, 102
ODCriteria.getFixedValues 102
ODCriteria.getName 94
ODCriteria.getOperand 94, 100
ODCriteria.getType 102
ODCriteria.getValidOperands 102
ODCriteria.getValues 102
ODCriteria.setOperand 102, 117
ODCriteria.setSearchValue 94, 117

ODCriteria.setSearchValues 94, 100, 102
oddělovací znaky 28
oddělovací znaky výstupu 28
oddělovaný výstup ASCII
 oddělovací znaky 28
ODFolder
 dokument, načtení 109
 dokument, tisk 112
 hledání 94, 98, 100, 102, 107, 109
 jméno 94, 107
 kritéria 94, 100, 102
 načtení dokumentu 109
 popis 94
 pořadí zobrazení 94, 107
 skupiny aplikací 92
 tisk dokumentů 112
 zavírání 92, 94, 98, 100
 zpráva 94
 zrušení hledání 100
ODFolder.close 92, 94, 98, 100, 109
ODFolder.getApplGroups 92
ODFolder.getCriteria 94, 100, 102
ODFolder.getDescription 94
ODFolder.getDisplayOrder 94, 98, 107
ODFolder.getName 94, 107
ODFolder.getNumApplGroups 92
ODFolder.getSearchMessage 94
ODFolder.printDocs 112
ODFolder.retrieve 109
ODFolder.search 94, 98, 100, 107, 109
ODFolder.setApplGroupForSearchWithSQL 98
ODHit
 aktualizace dokumentů 117
 anotace 113, 115
 dokument, aktualizace 117
 dokument, načtení 109
 hodnota zobrazení 107
 načtení dokumentu 109
 poznámky 113, 115
 seznam dokumentů 107
 typ dokumenty 94
 typ obsahu MIME 94
 umístění dokumentu 94
 zobrazovací hodnoty 94, 98
ODHit.addNote 115
ODHit.getDisplayValue 94, 98, 107, 117
ODHit.getDisplayValues 94
ODHit.getDocId 94, 109
ODHit.getDocLocation 94
ODHit.getDocType 94
ODHit.getMimeType 94
ODHit.getNotes 113, 115
ODHit.retrieve 109
ODHit.update 117
odhlášení
 API 63
 parametry 63
 popis funkce 6
 vzorové volání funkce 64
odkazy 33
ODNote
 anotace 113, 115
 barva 113
 čas 113
 datum 113
 ID uživatele 113
 jméno skupiny 113

- ODNote (*pokračování*)
 - pozice 113
 - strana 113
 - text 113
- ODNote.getColor 113
- ODNote.getDateTime 113
- ODNote.getGroupName 113
- ODNote.getOffsetX 113
- ODNote.getOffsetY 113
- ODNote.getPageNum 113
- ODNote.getText 113
- ODNote.getUserId 113
- ODNote.isOkToCopy 113, 115
- ODNote.isPublic 113, 115
- ODNote.setGroupName 115
- ODNote.setText 115
- odpojování od serveru 90
- ODServer
 - dokument, načtení 109
 - heslo 90, 120
 - ID uživatele 90
 - jméno aplikace 90
 - jméno složky 105
 - lokální adresář 90
 - načtení dokumentu 109
 - nastavení a získání hesla 90
 - nastavení a získání ID uživatele 90
 - nastavení hesel 120
 - odpojuje se od 90
 - otevírání složky 102
 - otevření složky 109
 - popis složky 105
 - port 90
 - připojování k serveru 89
 - připojuje se k 90
 - server 90
 - serverové tiskárny 112
 - složka, otevření 109
 - tiskárny 112
 - typ připojení 90
 - změna hesel 120
 - zrušení hledání 100
- ODServer.cancel 100
- ODServer.getConnectType 90
- ODServer.getFolderNames 105
- ODServer.getFoldersDescription 105
- ODServer.getLocalDir 90
- ODServer.getNumFolders 105
- ODServer.getPassword 90
- ODServer.getPort 90
- ODServer.getServerName 90
- ODServer.getServerPrinters 112
- ODServer.getUserId 90
- ODServer.changePassword 120
- ODServer.logoff 89, 90
- ODServer.logon 89, 90
- ODServer.openFolder 102, 109
- ODServer.retrieve 109
- ODServer.setApplicationName 90
- ODServer.setConnectType 90
- ODServer.setLocalDir 90
- ODServer.setPassword 90
- ODServer.setPort 90
- ODServer.setServer 90
- ODServer.setUserId 90
- ODServer.terminate 89, 90

- OnDemand Internet Connection
 - o aplikaci 1
- operandy, Java API 94
- oprávnění 9
- oprávnění k adresářům 9
- OS/400
 - instalace 10

P

- parametr ADDEXTENSION 29
- parametr ADDFIELDSTODOCID 29
- parametr ADDNOTES 29
- parametr AFPVIEWING 30
- parametr APPLETCACHEDIR 13
- parametr APPLETDIR 13
- parametr AUTODOCRETRIEVAL 30
- parametr BEGIN 28
- parametr CACHEDIR 14
- parametr CACHEDOCS 14
- parametr CACHEMAXTHRESHOLD 14
- parametr CACHEMINTHRESHOLD 15
- parametr CACHESIZE 15
- parametr CACHEUSERIDS 15
- parametr CODEPAGE 16, 135
- parametr CONFIGFILE 20, 22
- parametr DOCSIZE 16
- parametr EMAILVIEWING 31
- parametr ENCRYPTCOOKIES 31
- parametr ENCRYPTURL 32
- parametr END 28
- parametr FOLDERDESC 32
- parametr HOST 12
- parametr IMAGEDIR 16
- parametr INSTALLDIR 21, 22
- parametr LANGUAGE 17, 135
- parametr LINEVIEWING 32
- parametr LOG 36, 137
- parametr LOGDIR 36
- parametr MAXHITS 33
- parametr NOLINKS 33
- parametr ODApplet.jre.path.IE 33
- parametr ODApplet.jre.path.NN 34
- parametr ODApplet.jre.version 34
- parametr ODApplet.version 34
- parametr PORT 11, 12
- parametr PROTOCOL 11, 12
- parametr REPORTSERVERTIMEOUT 19
- parametr SEPARATOR 28
- parametr SERVERACCESS 19
- parametr SERVERPRINT 34
- parametr SERVERPRINTERS 34
- parametr SHOWDOCLOCATION 34
- parametr ShowSearchString 18
- parametr TEMPDIR 18
- parametr TEMPLATEDIR 19
- parametr USEEXECUTABLE 21, 22
- parametr VIEWNOTES 35
- parametry
 - ADDEXTENSION 29
 - ADDFIELDSTODOCID 29
 - ADDNOTES 29
 - AFPVIEWING 30
 - APPLETCACHEDIR 13
 - APPLETDIR 13
 - AUTODOCRETRIEVAL 30
 - BEGIN 28

- parametry (pokračování)
 - CACHEDIR 14
 - CACHEDOCS 14
 - CACHEMAXTHRESHOLD 14
 - CACHEMINTHRESHOLD 15
 - CACHESIZE 15
 - CACHEUSERIDS 15
 - CODEPAGE 16
 - CONFIGFILE 20, 22
 - DOCSIZE 16
 - EMAILVIEWING 31
 - ENCRYPTCOOKIES 31
 - ENCRYPTURL 32
 - END 28
 - FOLDERDESC 32
 - HOST 12
 - IMAGEDIR 16
 - INSTALLDIR 21, 22
 - LANGUAGE 17
 - LINEVIEWING 32
 - LOG 36, 137
 - LOGDIR 36
 - MAXHITS 33
 - NOLINKS 33
 - ODApplet.jre.path.IE 33, 34
 - ODApplet.jre.path.NN 34
 - ODApplet.version 34
 - PORT 11, 12
 - PROTOCOL 11, 12
 - REPORTSERVERTIMEOUT 19
 - sekce @SRV@_DEFAULT 11
 - sekce @SRV@_server 12
 - sekce AFP2HTML 20
 - sekce AFP2PDF 21
 - sekce ATTACHMENT IMAGES 26
 - sekce CONFIGURATION 13
 - sekce SECURITY 19
 - SEPARATOR 28
 - SERVERACCESS 19
 - SERVERPRINT 34
 - SERVERPRINTERS 34
 - SHOWDOCLOCATION 34
 - ShowSearchString 18
 - TEMPDIR 18
 - TEMPLATEDIR 19
 - USEEXECUTABLE 21, 22
 - VIEWNOTES 35
 - podpora národního jazyka. 135
 - popis složky, Java API 105
 - port
 - Java API 90
 - poznámky 29, 35
 - požadavky
 - AFP2PDF Transform 9
 - AFP2WEB Transform 9
 - cookie 42
 - HTTP server 9
 - Java API 1
 - java applet AFP2HTML 9
 - Java Virtual Machine 42
 - rychlá vyrovnávací paměť pro dokumenty 9
 - server 9
 - uložení do paměti cache 9
 - práva 9
 - programy typu plugin
 - instalace 41
 - o aplikaci 1
 - prohlížeč Line Data Viewer
 - konfigurace 48
 - ODApplet.version parameter 34
 - parametr ODApplet.jre.path.IE 33
 - parametr ODApplet.jre.path.NN 34
 - parametr ODApplet.jre.version 34
 - prohlížeče
 - cookie 42
 - Java Virtual Machine 42
 - JVM 42
 - podporované 42
 - proměnné prostředí, rozhraní Java API 86
 - prostředí systému, rozhraní Java API 86
 - protokoly 11, 12
 - průvodce programováním
 - API 85
 - Java API 85
 - přehled 1
 - přidání anotace
 - API 54
 - parametry 54
 - popis funkce 5
 - vzorové volání funkce 56
 - přihlášení
 - API 65
 - parametry 65
 - popis funkce 6
 - výstup ASCII s oddělovači 131
 - vzorové volání funkce 66
 - příklady 85
 - přílohy 26, 27
 - přílohy BMP 27
 - přílohy GIF 27
 - přílohy TXT 27
 - připojování k serveru 89
 - připojování k serveru 90
 - příprava na používání aplikace OnDemand Internet Connection 1
 - přístupový seznam serveru 19
- ## R
- reference
 - API 53, 81, 83
 - CGI API 53
 - Java API 83
 - Java servlet 81
 - servlet 81
 - rozhraní API
 - balení 85
 - proměnné prostředí 86
 - prostředí systému 86
 - třídy 85
- ## Ř
- řetězec hledání SQL s Java API 98
- ## S
- sekce @SRV@_DEFAULT 11
 - sekce @SRV@_server 12
 - sekce AFP2HTML 20
 - sekce AFP2PDF 21
 - sekce ATTACHMENT IMAGES 26
 - sekce browser 35
 - sekce CONFIGURATION 13

- sekce debug 36
 - sekce DEFAULT BROWSER 28
 - sekce MIMETYPES 22
 - sekce NOHTML 28
 - sekce SECURITY 19
 - server
 - Java API 90
 - serverový tisk
 - Java API 112
 - parametry 67
 - povolení 34
 - vzorové volání funkce 70
 - servlet
 - reference 81
 - seznam nalezených dokumentů
 - API 59
 - Java API 94, 98, 107, 109
 - parametry 59
 - popis funkce 5
 - výstup ASCII s oddělovači 133
 - vzorové volání funkce 62
 - skupiny aplikací ve složce
 - Java API 92
 - složka, hledání s Java API 94, 98, 100, 102, 107
 - složka, výpis skupin aplikací v rozhraní Java API 92
 - software ODWEK
 - instalace 9
 - soubor ARSWWW.INI (*pokračování*)
 - parametr PROTOCOL 11, 12
 - parametr REPORTSERVERTIMEOUT 19
 - parametr SEPARATOR 28
 - parametr SERVERACCESS 19
 - parametr SERVERPRINT 34
 - parametr SERVERPRINTERS 34
 - parametr SHOWDOCLOCATION 34
 - parametr ShowSearchString 18
 - parametr TEMPDIR 18
 - parametr TEMPLATEDIR 19
 - parametr USEEXECUTABLE 21, 22
 - parametr VIEWNOTES 35
 - sekce @SRV@_DEFAULT 11
 - sekce @SRV@_server 12
 - sekce AFP2HTML 20
 - sekce AFP2PDF 21
 - sekce ATTACHMENT IMAGES 26
 - sekce browser 35
 - sekce CONFIGURATION 13
 - sekce debug 36
 - sekce DEFAULT BROWSER 28
 - sekce MIMETYPES 22
 - sekce NOHTML 28
 - sekce SECURITY 19
 - volby prohlížeče 35
 - zadání 10
 - soubor httpd.conf 129
 - soubor šablony 40
 - soubory protokolu 36, 137
- ## Š
- šifrování 31, 32
- ## T
- TEMPLATE.HTM 40
 - tisk
 - Java API 112
 - tisk dokumentu
 - API 67
 - Java API 112
 - parametry 67
 - vzorové volání funkce 70
 - tisk dokumentů
 - popis funkce 6
 - tisk na serveru
 - API 67
 - popis funkce 6
 - tisknutí
 - server 34
 - trasování a diagnostické informace 88
 - trasování problémů 137
 - třídy 85
 - typ dokumentu, Java API 94
 - typ obsahu MIME 22, 94
 - typ připojení
 - Java API 90
 - typ/podtyp médií 22
- ## U
- ukládání dokumentů 14
 - uložení do paměti cache 14, 15
 - umístění dokumentu 34

určování problémů 137
uživatelsky definované soubory
instalace 43

V

velikost paměti cache 15
velké objekty 20, 125
volby HTTP server
parametr CACHEDIR 14
volby HTTP serveru
AFP2PDF Transform 21
AFP2WEB Transform 20
debug 36
parametr APPLETDIR 13
parametr BEGIN 28
parametr CACHEDOCS 14
parametr CACHEMAXTHRESHOLD 14
parametr CACHEMINTHRESHOLD 15
parametr CACHESIZE 15
parametr CACHEUSERIDS 15
parametr CODEPAGE 16
parametr CONFIGFILE 20, 22
parametr END 28
parametr IMAGEDIR 16
parametr INSTALLDIR 21, 22
parametr LANGUAGE 17
parametr REPORTSERVERTIMEOUT 19
parametr SEPARATOR 28
parametr SERVERACCESS 19
parametr TEMPDIR 18
parametr TEMPLATEDIR 19
parametr USEEXECUTABLE 21, 22
prohlížeče 28, 35
předvolený prohlížeč 28
sekce AFP2HTML 20
sekce AFP2PDF 21
sekce ATTACHMENT IMAGES 26
sekce CONFIGURATION 13
sekce MIMETYPES 22
sekce NOHTML 28
sekce SECURITY 19
volby pro HTTP server
parametr ShowSearchString 18
volby pro server OnDemand
parametr HOST 12
parametr PORT 11, 12
parametr PROTOCOL 11, 12
parametry 12
předvolby 11
sekce @SRV@_DEFAULT 11
sekce @SRV@_server 12
volby prohlížeče
sekce browser 35
sekce DEFAULT BROWSER 28
volby předvoleného prohlížeče 28
volby webového serveru
DBCS 135
jazyk 135
kódová stránka 135
NLS 135
vyhledávací kritéria
API 74
Java API 94, 98, 102
parametry 74
popis funkce 6
řetězec SQL 98

vyhledávací kritéria (*pokračování*)
výstup ASCII s oddělovači 132
vzorové volání funkce 75
výsledky dotazu 33
výstup ASCII
anotace 134
formát 131
generovaný na OnDemand 131
chybová zpráva 134
přihlášení 131
seznam nalezených dokumentů 133
vyhledávací kritéria 132
zobrazení anotací 134
zprávy 134
výstup ASCII s oddělovači
anotace 134
formát 131
generovaný na OnDemand 131
chybová zpráva 134
přihlášení 131
seznam nalezených dokumentů 133
vyhledávací kritéria 132
zobrazení anotací 134
zprávy 134
vzorové aplikace 39
vzorový kód 85

W

webové aplikace
vzory 39
webové stránky
vzory 39

Z

zabezpečení dat 6
zabezpečení ochrany dat 6, 19, 31, 32
zabezpečení serveru 6, 19
zacházení s výjimkami 88
zápis do protokolu 36, 137
změna hesel 120
změna hesla
API 57
parametry 57
popis funkce 5
vzorové volání funkce 58
zobrazení anotací
API 78
parametry 78
popis funkce 6
výstup ASCII s oddělovači 134
vzorové volání funkce 79
zobrazení umístění dokumentu 34
zobrazovací hodnoty, Java API 94
zprávy 17
výstup ASCII s oddělovači 134
zrušení hledání 100

Ž

žádný HTML výstup 131
žádný výstup HTML 28



Číslo programu: 5722-RD1

SC09-3699-02

