WebSphere Product Center: Installation Guide

Version 5.1

Note! Before using this information and the product it supports, read the information in "Notices" at the end of this document.

25 September2004

This edition of this document applies to WebSphere Product Center (5724-I68), version 5.1, and to all subsequent releases and modifications until otherwise indicated in new editions.

Table of Contents

Ch 1 - Introduction	1-1
Audience	1-1
Document Structure	1-1
Related documentation	1-1
Typographic conventions	1-2
Ch 2 - Overview	2-1
Architecture	2-2
Information flow and HA configurations	2-3
Ch 3 - Installation requirements	3-1
Installation requirements	3-1
AIX requirements	3-2
Linux requirements	3-4
HP-UX requirements	3-6
Requirements for client access to WebSphere Product Center	3-7
Team requirements	3-8
Ch 4 - Database setup and configuration	4-1
DB2 setup and configuration	4-1
Oracle setup and configuration	4-22
Ch 5 WebSphere Product Center installation	5-1
Satisfy prerequisites	5-1
Preparing the installation media	5-1
General procedures for using installation media	5-1
Installing WebSphere Product Center	5-4
Set environment variables	5-5
Configure run time properties	5-9
Test for valid license file	5-12
Application server configuration	5-13
Setup database schema	5-21
Setup a company to test	5-23
Test installation	5-24
Create a new company	5-28
Ch 6 - Uninstalling WebSphere Product Center	6-1
Invoking the graphical uninstaller	6-1
Using the graphical uninstaller	6-2

Performing a silent uninstallation	6-2
Ch 7 - Administrating WebSphere Product Center services	7-1
Service types	7-1
Appendix A - Configuration reference	A-1
File name: common.properties	A-1
File name: admin_properties.xml	A-28
File name: docstore_mount.xml	A-29
File name: data_entry_properties.xml	A-30
Notices	

Ch 1 - Introduction

Audience

This document is for consultants, developers, and system administrators who plan, install, deploy, and administer WebSphere Product Center 5.0.1.

Document structure

Chapter	Title	Description
1	Overview	Brief architecture and feature overview
2	Installation for AIX	Installation requirements for AIX
3	Installation for Linux	Installation requirements for Linux
4	Uninstalling WebSphere Product Center	Uninstall instructions
5	Administration	Administration of WebSphere Product Center
Appendix A		List of configuration properties

Related documentation

The following related WebSphere Product Center documentation is also available on IBM's internet site

(http://www-306.ibm.com/software/integration/wpc/library/):

- WebSphere Product Center: Support Guide
- WebSphere Product Center: Release Notes for Version 5.0.1

Typographic conventions

This document uses the following conventions:

`Courier New`	Indicates commands, directory names, user names, path names, and file names.
`Courier New`	Indicates an example
bold	Indicates an action to be performed in a GUI

Ch 2 Overview

WebSphere Product Center provides a highly scalable, enterprise Product Information Management (PIM) solution. Product Information Management is the middleware that establishes a single, integrated, consistent view of product or services information inside and outside an enterprise. As customers become global, on demand e-businesses, a golden source of product and services information is needed to address business needs such as global data synchronization, ecommerce, supply chain management and trading partner management.

Using Product Information Management middleware, companies can manage, link and synchronize item, location, organization, trading partner and trade terms internally and externally. Product information management solution using WebSphere Product Center provides:

- A flexible, scalable repository managing and linking product, location, trading partner, organization, and terms of trade information
- Tools for modeling, capturing, creating and managing this information with high user productivity and high information quality
- Integrating and synchronizing this information internally with legacy systems, enterprise applications, repositories and masters
- Business user workflow for supporting multi-department and multi-enterprise business processes
- Exchanging and synchronizing this information externally with business partners
- Leveraging this information via many internal and external electronic and human touch points

Using WebSphere Product Center's enterprise solution, companies can dramatically accelerate the time to market for new products, increase market share through tighter collaboration with trading partners and improve customer satisfaction, while reducing costs.

Architecture

WebSphere Product Center's scalable architecture allows the ability to be easily implemented in a production environment that provides security, data archival, and operational redundancy. Figure 1 illustrates a two-tier architecture and Figure 2 illustrates a three-tier architecture.

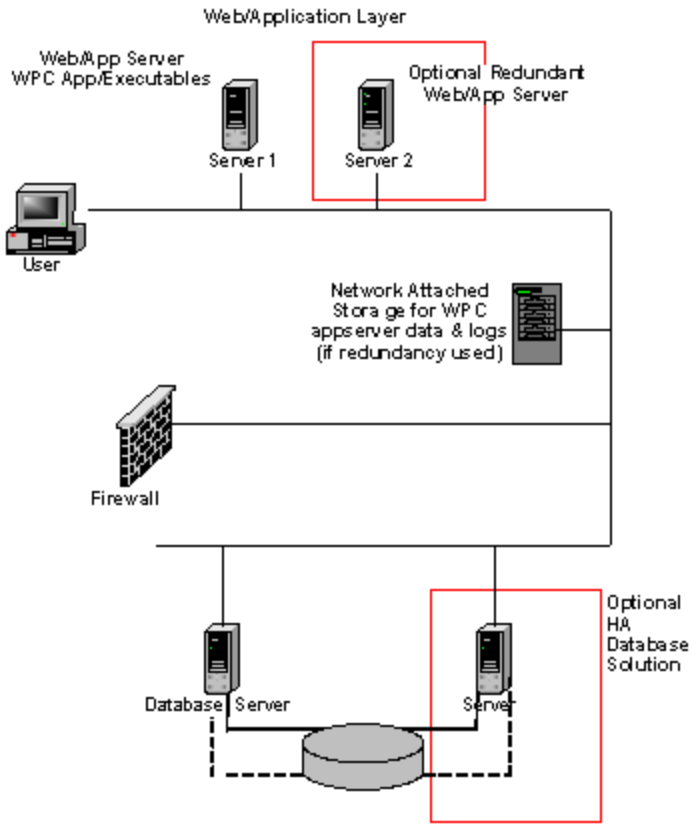

Figure 1 - Two-tier architecture

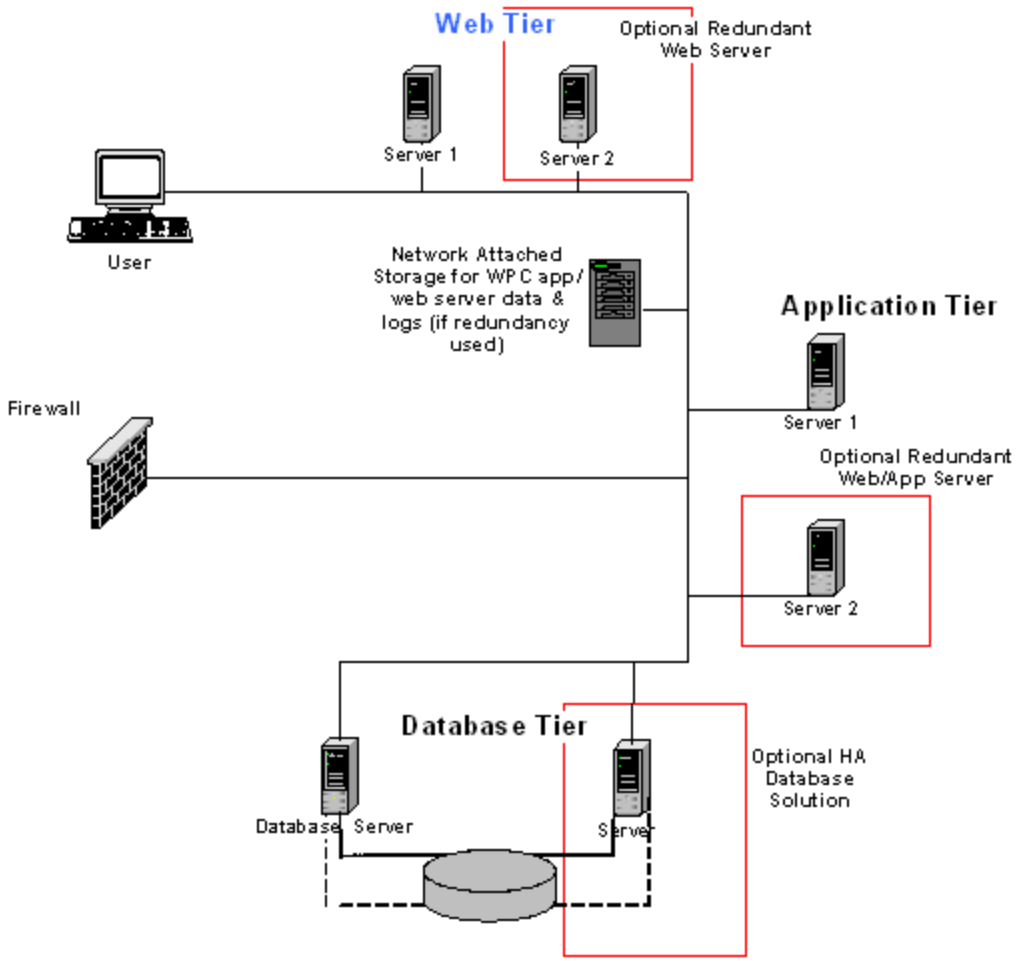

Figure 2 - Three-tier architecture

Information Flow and High Availability Configurations

The flow of information is interdependent on several components. Figure 3 provides an example of the flow of the security and information between the client, web/application server (WebSphere Product Center installed), and the database. The scheduler service can be on the application server or on a separate server, based on load requirements. If the scheduler is placed on a separate server, WebSphere Product Center can bind the scheduler service to a specific RMI (Remote Method Invocation) port.

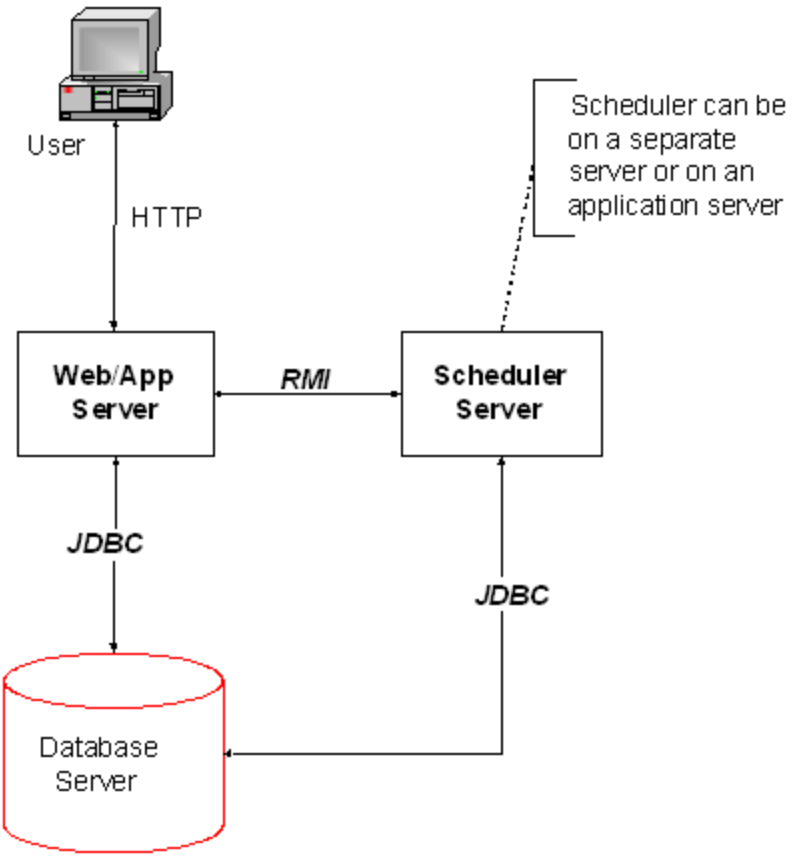

Figure 3 – Flow of information

Summary

The following list is the result of information flow illustrated in Figure 3.

- The application server itself handles HTTP requests from user
- RMI on the application server is used to control services
- The Scheduler server uses the same RMI port used by the RMI used to control services
- Application and scheduler servers communicate with the database server using JDBC
- The scheduler can be run on a dedicated machine or on an application server

Ch 3 Installation requirements

Before you install the IBM WebSphere Product Center software, ensure that you have all the necessary prerequisites outlined in this chapter. The sections in this chapter provide a brief overview of the system hardware and software requirements, team requirements, and the application server/database configurations required to run WebSphere Product Center.

For the best performance, it is recommended to run WebSphere Product Center on a dedicated system. The system should have restricted access to maintain security.

Hardware

Recommendations for hardware requirements are supplied for guidance only. This section lists the minimum hardware requirements for WebSphere Product Center. However, the actual requirements for your system may be greater, depending on the complexity of your specific WebSphere Product Center environment, throughput, and data object size. The following information is for the WebSphere Product Center system only. If you run other applications on the same system, make appropriate adjustments.

Additional server hardware requirements

1. Additional CPUs may be required based on capacity requirements

2. Additional memory may be required based on capacity requirements

3. Additional servers for added capacity and/or redundancy

4. Multi-server installations require network attached shared storage

Software

The software requirements for WebSphere Product Center depend on which of the elements of the product components are being used and whether singly or in combination on a particular server. Dedicated

servers are recommended for the database.

AIX requirements

This section outlines the hardware and software requirements for the following platform:

Operating System	Application Server	Database Server
AIX 5L 5.2 (32 & 64 bit)	WebSphere Application Server Network Deployment 5.0.2.5	IBM DB2 II Version 8.1 Advanced Edition, Fix Pack 5 OR Oracle 9.2.0.5 Enterprise Edition

WebSphere Application Server/DB2 on AIX

Note: This is the primary platform used for WebSphere Product Center

Application Server	
Software	• WebSphere Application Server Network Deployment 5.0.2.5 • AIX 5L 5.2 (32 & 64 bit) • IBM HTTP Server 2.0 • IBM JDK bundled with WebSphere Application Server Network Deployment 5.0.2.5
Hardware	• Server Type – IBM p630 Model 6C4 • Processor – 4 x 64-bit POWER4+ 1.45 GHz • Memory – 8 GB • Disk Sub-System – 144 GB SCSI
Database Server	
Software	• DB2 II 8.1 Advanced Edition Fixpack 5 • AIX 5L 5.2

Hardware	• Server Type – IBM p630 Model 6C4, PE2650 • Processor – 4 x 64-bit POWER4+ 1.45 GHz • Memory – 8 GB • Disk Sub-System – 420 GB SCSI
Additional Software	
This primary stack is also bundled with the following: • WebSphere MQ Series V5.3 (1) • WebSphere Business Integration Adapter for WebSphere MQ V2.5.0 (1, 2) • WebSphere Business Integration Adapter Framework V2.4.0 (1, 2) (1) A limited use license is included with WebSphere Product Center (2) Required when integrating with a process hub	

WebSphere Application Server/Oracle on AIX

Application Server	
Software	• WebSphere Application Server Network Deployment 5.0.2.5 • AIX 5L 5.2 • IBM HTTP Server 2.0 • IBM JDK bundled with WebSphere Application Server Network Deployment 5.0.2.5
Hardware	• Server Type – IBM p630 Model 6C4 • Processor – 4 x 64-bit POWER4+ 1.45 GHz • Memory – 8 GB • Disk Sub-System – 144 GB SCSI
Database Server	
Software	• Oracle 9.2.0.5 Enterprise Edition • AIX 5L 5.2
	• Server Type: Intel compatible • Processor: Dual Intel, XEON 2.8 GHz

Hardware	• Memory: 6 GB • Disk Sub-System: 365 GB 10K RPM SCSI Driver

Linux requirements

Operating System	Application Server	Database Server
Red Hat Enterprise Linux ES 2.1	WebSphere Application Server 5.0.2.5	IBM DB2 II Version 8.1 Advanced Edition, Fix Pack 5 OR Oracle 9.2.0.5 Enterprise Edition
Red Hat Enterprise Linux ES 2.1	WebLogic Server 7.0	Oracle 9.2.0.5 Enterprise Edition

WebSphere Application Server/DB2 on Linux

Application Server	
Software	• WebSphere Application Server Network Deployment 5.0.2.5 • IBM HTTP Server 2.0 • Red Hat Enterprise Linux ES 2.1 • IBM JDK bundled with WebSphere Application Server Network Deployment
Hardware	• Server Type: Intel compatible • Processor: Dual Intel, XEON 2.8 GHz • Memory: 6 GB • Disk Sub-System: 365 GB 10K RPM SCSI Driver
Database Server	

Software	• DB2 II 8.1 Advanced Edition Fixpack 5 • Red Hat Enterprise Linux ES 2.1
Hardware	• Server Type: Intel compatible • Processor: Dual Intel, XEON 2.8 GHz • Memory: 6 GB • Disk Sub-System: 365 GB 10K RPM SCSI Driver

WebSphere Application Server/Oracle on Linux

Application Server	
Software	• WebSphere Application Server Network Deployment 5.0.2.5 • IBM HTTP Server 2.0 • Red Hat Enterprise Linux ES 2.1 • IBM JDK bundled with WebSphere Application Server Network Deployment
Hardware	• Server Type: Intel compatible • Processor: Dual Intel, XEON 2.8 GHz • Memory: 6 GB • Disk Sub-System: 365 GB 10K RPM SCSI Driver
Database Server	
Software	• Oracle 9.2.0.5 Enterprise Edition • Red Hat Enterprise Linux ES 2.1
Hardware	• Server Type: Intel compatible • Processor: Dual Intel, XEON 2.8 GHz • Memory: 6 GB • Disk Sub-System: 365 GB 10K RPM SCSI Driver

WebLogic/Oracle on Linux

Application Server	
	• WebLogic Server 7.0

Software	• Apache HTTP Server 2.0 • Red Hat Enterprise Linux ES 2.1 • WebLogic JDK bundled with WebLogic Server 7.0
Hardware	• Server Type: Intel compatible • Processor: Dual Intel, XEON 2.8 GHz • Memory: 6 GB • Disk Sub-System: 365 GB 10K RPM SCSI Driver
Database Server	
Software	• Oracle 9.2.0.5 Enterprise Edition • Red Hat Enterprise Linux ES 2.1
Hardware	• Server Type: Intel compatible • Processor: Dual Intel, XEON 2.8 GHz • Memory: 6 GB • Disk Sub-System: 365 GB 10K RPM SCSI Driver

HP-UX requirements

Operating System	Application Server	Database Server
HP UX 11i	WebLogic Server 7.0	Oracle 9.2.0.5 Enterprise Edition

WebLogic/Oracle on HP-UX

Application Server	
Software	• WebLogic Server 7.0 • Apache HTTP Server 2.0 • HP/UX 11i • WebLogic JDK bundled with WebLogic Server 7.0
	• Server Type: HP RP 4440 • Processor: 2 CPUs

Hardware	• Memory: 4 GB • Disk Sub-System: HP MSA 1000
Database Server	
Software	• Oracle 9.2.0.5 Enterprise Edition • HP/UX 11i
Hardware	• Server Type: HP RP 4440 • Processor: 2 CPUs • Memory: 4 GB • Disk Sub-System: HP MSA 1000

Requirements for client access to WebSphere Product Center

The following represents the minimum client hardware requirements for client access to WebSphere Product Center:

Client requirements	
Software	Microsoft Windows 2000 Professional Service Pack 3 OR Microsoft Windows XP Professional Service Pack 1 AND Browser: Microsoft Internet Explorer Version 6.0 Service Pack 1
Hardware	Processor: Intel-based system with a Pentium 3 processor and a minimum processor speed of 750 MHz Memory: 256 MB

Team Requirements

The following personnel are needed for the installation and maintenance of WebSphere Product Center.

- Project Manager
- Database Administrator
- WebSphere Product Center Middleware Administrator
- System Administrator (may be the same as the WebSphere Product Center Middleware Administrator)
- Network Operations
- Security/Firewall Administrator

Ch 4 Database setup and configuration

DB2 setup and configuration

This section covers the database configuration guidelines that are used by a database administrator to create and configure a DB2 database for WebSphere Product Center.

Satisfy prerequisites

- Install IBM DB2 II Version 8.1 Advanced Edition or IBM DB2 UDB 8.1 Enterprise Server Edition on AIX Version 5L v5.1
- Apply DB2 Fix Pack 5 on the database server

When the above prerequisites have been satisfied, use the following guidelines outlined in this section to create and setup the WebSphere Product Center database.

Create an instance for WebSphere Product Center database

Create an Instance with the name **db2inst1** for the WebSphere Product Center database. If desired, create a 64-bit instance for the database. An option is provided to create an instance at the time of DB2 software installation.

Create new database

Sharing an existing database on a machine with WebSphere Product Center Middleware Database is not recommended. Create a new database for WebSphere Product Center Schema. You can create the database, bufferpools and tablespaces required at the time of DB2 Software Installation.

Note: The database MUST be created using CODESET UTF-8. The

database name used in the examples in this guide is **WPCDB**.

Create new bufferpools

Due to the large size of tables in WebSphere Product Center, the page size used to create the bufferpools is 16KB.

The following bufferpools are needed for use by tablespaces.

- USERSBP – For use by USERS tablespace
- INDXBP – For use by INDX tablespace
- BLOBBP – For use by BLOB_TBL_DATA tablespace
- TEMPUSRBP – For use by user's temporary tablespace
- TEMPSYSBP – For use by system's temporary tablespace

The following table lists the recommended size for each bufferpool.

Bufferpool	Size (16KB Pages)*
USERSBP	30000
INDXBP	30000
BLOBBP	1000
TEMPUSRBP	10000
TEMPSYSBP	10000

* These figures were calibrated for a server using 4 GB of memory. The sizes can be increased with 0.25 times per GB increase on the database server memory.

* It is necessary that the instance be restarted to activate the bufferpools using db2stop and db2start commands before creating the new tablespaces.

Sample SQL as created by Control Center:

```
db2 CONNECT TO wpcdb
db2 CREATE BUFFERPOOL USERSBP SIZE 30000 PAGESIZE 16384
db2 CREATE BUFFERPOOL INDXBP SIZE 30000 PAGESIZE 16384
db2 CREATE BUFFERPOOL BLOBBP SIZE 1000 PAGESIZE 16384
```

db2 CREATE BUFFERPOOL TEMPUSRBP SIZE 10000 PAGESIZE 16384
db2 CREATE BUFFERPOOL TEMPSYSBP SIZE 10000 PAGESIZE 16384

Create new tablespaces

Due to the large size of tables in WebSphere Product Center the page size used to create the tablespaces is 16KB.

WebSphere Product Center requires the following tablespaces.

- USERS
- INDX
- BLOB_TBL_DATA
- TEMP_USER
- TEMP_SYSTEM

The following table lists the type, bufferpool and the node group assignment for the tablespaces.

Tablespace	Type	Management	Bufferpool
USERS	REGULAR	Database	USERSBP
INDX	REGULAR	Database	INDXBP
BLOB_TBL_DATA	REGULAR	Database	BLOBBP
TEMP_USER	USER TEMPORARY	System	TEMPUSRBP
TEMP_SYSTEM	SYSTEM TEMPORARY	System	TEMPSYSBP

Note: If Database Managed Tablespaces are put to use, ensure that enough containers are created and allocated for each tablespace. Please make sure that TEMP_USER is created as USER TEMPORARY tablespace and TEMP_SYSTEM is created as SYSTEM TEMPORARY

tablespace and both are managed by the system.

Sample SQL as created by Control Center:

Note: Directory path used to create tablespaces in the followng examples is /u01/db2data/wpcdb/

```
db2 CONNECT TO wpcdb;
db2 CREATE REGULAR TABLESPACE USERS PAGESIZE 16K
MANAGED BY DATABASE
USING (file '/u01/db2data/wpcdb/users01' 90000)
EXTENTSIZE 32 PREFETCHSIZE 64 BUFFERPOOL USERSBP
OVERHEAD 24.100000 TRANSFERRATE 0.900000 DROPPED TABLE
RECOVERY ON;

db2 CREATE REGULAR TABLESPACE INDX PAGESIZE 16K
MANAGED BY DATABASE
USING (file '/u01/db2data/wpcdb/indx01' 190000)
EXTENTSIZE 32 PREFETCHSIZE 64 BUFFERPOOL INDXBP
OVERHEAD 24.100000 TRANSFERRATE 0.900000 DROPPED TABLE
RECOVERY ON;

db2 CREATE REGULAR TABLESPACE BLOB_TBL_DATA
PAGESIZE 16K MANAGED BY DATABASE
USING (file '/u01/db2data/wpcdb/blob01' 60000)
EXTENTSIZE 32 PREFETCHSIZE 64 BUFFERPOOL BLOBBP
OVERHEAD 24.100000 TRANSFERRATE 0.900000 DROPPED TABLE
RECOVERY ON;

db2 CREATE USER TEMPORARY TABLESPACE TEMP_USER
PAGESIZE 16K MANAGED BY SYSTEM
USING ('/u01/db2data/wpcdb/usertemp01')
EXTENTSIZE 32 PREFETCHSIZE 64 BUFFERPOOL TEMPUSRBP
OVERHEAD 24.100000 TRANSFERRATE 0.900000;

db2 CREATE SYSTEM TEMPORARY TABLESPACE TEMP_SYSTEM
PAGESIZE 16K MANAGED BY SYSTEM
USING ('/u01/db2data/wpcdb/systemtemp01')
EXTENTSIZE 32 PREFETCHSIZE 64 BUFFERPOOL TEMPSYSBP
OVERHEAD 24.100000 TRANSFERRATE 0.900000;
```

Create AIX user

The WebSphere Product Center database schema needs a database user with user authentication done at the server level.

- Create the O/S user with the name **WPC** at the O/S level for use by the WebSphere Product Center database schema.
- Set the password of the user at the O/S level and try to connect to the user once at the O/S level to verify that the user can connect to the server.
- Set the primary group to **db2iadm1** or **db2grp1** in AIX as created.

Note: It is recommended to use the SMIT Administration Utility of AIX 5L to create a new AIX user.

Add database user and grant permissions

After creating the user at the O/S level, create the database user **WPC** and give the user the following permissions using the instance owner login (default instance owner login is **db2inst1**):

- DBADM
- CREATETAB
- BINDADD
- CONNECT
- CREATE_NOT_FENCED
- IMPLICIT_SCHEMA
- LOAD ON DATABASE

  Sample SQL as created by Control Center:

  db2 CONNECT TO wpcdb

  GRANT DBADM, CREATETAB, BINDADD, CONNECT, CREATE_NOT_FENCED, IMPLICIT_SCHEMA, LOAD ON DATABASE TO USER WPC;

In addition to this, grant permission to use space on all the WebSphere

Product Center specific tablespaces.

Sample SQL as created by Control Center:

GRANT USE OF TABLESPACE USERS TO WPC;
GRANT USE OF TABLESPACE INDX TO WPC;
GRANT USE OF TABLESPACE BLOB_TBL_DATA TO WPC;
GRANT USE OF TABLESPACE TEMP_USER TO WPC;

Create a new schema

Create a new schema WPC for user WPC.

Sample SQL as created by Control Center:

CREATE SCHEMA WPC AUTHORIZATION WPC;

Note: Repeat steps starting from “Create AIX user” in the previous section to “Create a new schema” if you want one more database schema user for another instance of WebSphere Product Center. For example if you want another test instance of WebSpehere Product Center running on the application server that needs a database schema in the same database then you need to create a database user and schema with the name **WPCTEST** in the database. This needs a O/S user with the name **WPCTEST**.

Catalog the node and the database on App Server

When running WebSphere Product Center and database on the different servers

When running WebSphere Product Center on a different server, catalog the database to be able to connect to the database from WebSphere Product Center. Run the following commands on the application server

db2 "catalog tcpip node <nodename> remote <dbhostname> server <sname/port#>"
db2 terminate
db2 "catalog database <dbname> as <dbname> at node <nodename>"
db2 terminate

Where:

- nodename - arbitrary name for the remote instance
- dbhostname - hostname or IP addr of db server
- sname/port# - service name or port# for the connection port of the local DB2 instance, in the /etc/services file.
- dbname – database name

Note: To identify the correct connection port see the value of the **SVCNAME** parameter of database manager (DBM).
Db2 get dbm cfg|grep “SVCNAME”

Example:

db2 "catalog tcpip node NODE0001 remote trigprd server 50000/tcp"
db2 terminate
db2 "catalog database wpcdb as wpcdb at node NODE0001"
db2 terminate

When running WebSphere Product Center and database on the same server

There is an issue when running WebSphere Product Center on the same box with DB2 on AIX. The following two fixes are required to make the WebSphere Product Center work.

1. Export EXTSHM=ON in the .profile file and .bashrc file of db2 database instance owner and the user used to install the WebSphere Product Center application.

```
export EXTSHM=ON
```

2. Set DB2ENVLIST with db2set command as db2 instance owner who created the database

```
db2set DB2ENVLIST=EXTSHM
```

Issue 'db2stop force' and 'db2start'

DB2 database profile registry updates

The following Profile Registry variables are required for use by WebSphere Product Center:

- DB2_RR_TO_RS
- DB2CODEPAGE
- DB2COMM

Other Profile Registry variables are not required, but may be set if there is a specific requirement.

DB2_RR_TO_RS

Description: Next key locking guarantees Repeatable Read (RR) isolation level by automatically locking the next key for all INSERT and DELETE statements and the next higher key value above the result set for SELECT statements.

For UPDATE statements that alter key parts of an index, the original index key is deleted and the new key value is inserted. Next key locking is done on both the key insertion and key deletion. The skipping behavior affects the RR, Read Stability (RS), and Cursor Stability (CS) isolation levels. (There is no row locking for Uncommitted Read (UR) isolation level.) When DB2_RR_TO_RS is on, RR behavior cannot be guaranteed for scans on user tables because next key locking is not done during index key insertion and deletion.

Catalog tables are not affected by this option. The other change in behavior is that with DB2_RR_TO_RS on, scans will skip over rows that have been deleted but not committed, even though the row may have qualified for the scan.

Value: Set it to ON

Example:

db2set db2_rr_to_rs=ON

DB2CODEPAGE

Description: Codepage is used to specify the character set that is used

during export and import of data in DB2. Set it to 1208

Value : Set it to 1208

Example:

db2set db2codepage=1208

DB2COMM

Description: The db2comm registry variable determines which protocol's connection managers will be enabled when the database manager is started. You can set this variable for multiple communication protocols by separating the keywords with commas.

Value: Set it to tcpip

Example:

db2set db2comm=tcpip

DB2 Database Manager configuration

The following Database Manager Configuration parameters have to be set for use with WebSphere Product Center.

- MON_HEAP_SZ - Database System Monitor Heap Size
- SHEAPTHRES - Sort Heap Threshold
- ASLHEAPSZ - Application Support Layer Heap Size
- QUERY_HEAP_SZ - Query Heap Size
- MAXAGENTS - Maximum Number of Agents

	Description	Value	Example
MON_HEAP_SZ	The memory required for maintaining the private views of the database system monitor data is allocated from the monitor heap. Its size is controlled by the mon_heap_sz configuration	Set it to 30000	

	parameter.		
SHEAPTHRES	Private and shared sorts use memory from two different memory sources. The size of the shared sort memory area is statically predetermined at the time of the first connection to a database based on the value of sheapthres. This has to at least two times the size of sortheap of any database hosted by the DB2 instance.	Set it to 20000	
ASLHEAPSZ	The application support layer heap represents a communication buffer between the local application and its associated agent. This buffer is allocated as shared memory by each database manager agent that is started.	Set it to 4200	
QUERY_HEAP_SZ	This parameter specifies the maximum amount of memory that can be allocated for the query heap. A query heap is used to store each query in the agent's private memory. As a minimum, you should set query_heap_sz to a value at least five times larger than aslheapsz.	Set it to 524280	
MAXAGENTS	This parameter indicates the maximum number of database manager agents, whether coordinating agents or subagents, available at any given time to accept WebSphere Product Center requests. The value of maxagents should be at least the sum of the values for maxappls in each database allowed to be accessed concurrently. If the number of databases is greater than	Set it to 400	Example Script : update dbm cfg using MON_HEAP_SZ 30000; update dbm cfg using SHEAPTHRES 20000; update dbm cfg using ASLHEAPSZ 4200; update dbm cfg using QUERY_HEAP_SZ 524280;

	the numdb parameter, then the safest course is to use the product of numdb with the largest value for maxappls.		update dbm cfg using MAXAGENTS 400;

DB2 database configuration parameters

The following Database Configuration parameters must be set for use with WebSphere Product Center.

- DFT_QUERYOPT - Default query optimization class
- DBHEAP - Database heap
- CATALOGCACHE_SZ - Catalog cache size
- LOGBUFSZ - Log buffer size
- UTIL_HEAP_SZ - Utilities heap size
- BUFFPAGE - Buffer pool size
- LOCKLIST - Maximum storage for lock list
- APP_CTL_HEAP_SZ - Maximum application control heap size
- SORTHEAP - Sort list heap
- STMTHEAP - SQL statement heap
- APPLHEAPSZ - Default application heap
- STAT_HEAP_SZ - Statistics heap size
- MAXLOCKS - Percentage of lock lists per application
- LOCKTIMEOUT - Lock timeout
- NUM_IOCLEANERS - Number of asynchronous page cleaners
- NUM_IOSERVERS - Number of I/O servers
- MAXAPPLS - Maximum number of active applications
- AVG_APPLS - Average number of active applications
- MAXFILOP - Maximum DB files open per application
- NEWLOGPATH – New path used to create the log files
- LOGFILSIZ - Log file size
- LOGPRIMARY - Number of primary log files
- LOGSECOND - Number of secondary log files

	Description	Value
	The query optimization class is used to direct the optimizer to use different	

DFT_QUERYOPT	degrees of optimization when compiling SQL queries. This parameter provides additional flexibility by setting the default query optimization class.	Set it to 9
DBHEAP	There is one database heap per database, and the database manager uses it on behalf of all instances of WebSphere Product Center connected to the database. It contains control block information for tables, indexes, table spaces, and buffer pools. It also contains space for the log buffer (logbufsz), and the catalog cache (catalogcache_sz). Therefore, the size of the heap will be dependent on the number of control blocks stored in the heap at a given time. The control block information is kept in the heap until all instances of WebSphere Product Center disconnect from the database.	Set it to 65448

	The minimum amount the database manager needs to get started is allocated at the first connection. The data area is expanded as needed up to the maximum specified by dbheap.	
CATALOGCACHE_SZ	This parameter indicates the maximum amount of space that the catalog cache can use from the database heap (dbheap).	Set it to 6000
LOGBUFSZ	This parameter allows you to specify the amount of the database heap (defined by the dbheap parameter) to use as a buffer for log records before writing these records to disk This parameter must also be less than or equal to the dbheap parameter.	Set it to 4096
UTIL_HEAP_SZ	This parameter indicates the maximum amount of memory that can be used simultaneously by the BACKUP, RESTORE and	Set it to 5000

	LOAD and load recovery utilities.	
BUFFPAGE	The buffpage parameter controls the size of a buffer pool when the CREATE BUFFERPOOL or ALTER BUFFERPOOL statement was run with NPAGES -1; otherwise, the buffpage parameter is ignored and the buffer pool will be created with the number of pages specified by the NPAGES parameter.	Set it to 22000
LOCKLIST	This parameter indicates the amount of storage that is allocated to the lock list. There is one lock list per database and it contains the locks held by all instances of WebSphere Product Center concurrently connected to the database. This parameter might require an increase depending the size of the database.	Set it to 6000
	This parameter determines the maximum size, in 4 KB pages, for the application	

APP_CTL_HEAP_SZ	control shared memory. Application control heaps are allocated from this shared memory.	Set it to 4500
SORTHEAP	This parameter defines the maximum number of private memory pages to be used for private sorts, or the maximum number of shared memory pages to be used for shared sorts.	Set it to 2650
STMTHEAP	The statement heap is used as a workspace for the SQL compiler during compilation of an SQL statement. This parameter specifies the size of this workspace.	Set it to 30000
APPLHEAPSZ	This parameter defines the number of private memory pages available to be used by the database manager on behalf of a specific agent or subagent.	Set it to 45000
STAT_HEAP_SZ	This parameter indicates the maximum size of the heap used in collecting statistics using the RUNSTATS command.	Set it to 22000

MAXLOCKS	Lock escalation is the process of replacing row locks with table locks, reducing the number of locks in the list. This parameter defines a percentage of the lock list held by an application that must be filled before the database manager performs escalation.	Set it to 30
LOCKTIMEOUT	This parameter specifies the number of seconds that WebSphere Product Center will wait to obtain a lock.	Set it to 8
NUM_IOCLEANERS	This parameter allows you to specify the number of asynchronous page cleaners for a database. These page cleaners write changed pages from the buffer pool to disk before a database agent requires the space in the buffer pool.	Set it to 7
NUM_IOSERVERS	I/O servers are used on behalf of the database agents to perform prefetch I/O and asynchronous I/O by utilities such as backup and restore. This	Set it to 8

	parameter specifies the number of I/O servers for a database.	
MAXAPPLS	This parameter specifies the maximum number of concurrent instances of WebSphere Product Center that can be connected (both local and remote) to a database.	Set it to 400
AVG_APPLS	The SQL optimizer to help estimate how much buffer pool will be available at run-time for the access plan chosen uses this parameter.	Set it to 2
		Set it to 640 Example Script: (Name of the database used is WPCDB) db2 connect to wpcdb update db cfg for wpcdb using DFT_QUERYOPT 9; update db cfg for wpcdb using DBHEAP 65448; update db cfg for wpcdb using CATALOGCACHE_SZ 6000; update db cfg for wpcdb using LOGBUFSZ 4096; update db cfg for wpcdb using UTIL_HEAP_SZ 5000; update db cfg for wpcdb using BUFFPAGE 22000; update db cfg for wpcdb using LOCKLIST 6000;

MAXFILOP	This parameter specifies the maximum number of file handles that can be open for each database agent.	6000; update db cfg for wpcdb using APP_CTL_HEAP_SZ 4500; update db cfg for wpcdb using SORTHEAP 2650; update db cfg for wpcdb using STMTHEAP 30000; update db cfg for wpcdb using APPLHEAPSZ 45000; update db cfg for wpcdb using STAT_HEAP_SZ 22000; update db cfg for wpcdb using MAXLOCKS 30; update db cfg for wpcdb using LOCKTIMEOUT 8; update db cfg for wpcdb using NUM_IOCLEANERS 7; update db cfg for wpcdb using NUM_IOSERVERS 8; update db cfg for wpcdb using MAXAPPLS 400; update db cfg for wpcdb using AVG_APPLS 2; update db cfg for wpcdb using MAXFILOP 640;

Setting up the transaction log files for WebSphere Product Center database

The log files exist to provide the ability to be able to recover your environment to a consistent state and preserve the integrity of your data. Log files storage needs to be optimize because they are not only written sequentially, the database manager will need to read the log files during database recovery.

It is recommended to put the logs on file system and they should always reside on their own physical disks, separate from the database tablespaces and database software. The disks ideally should be dedicated to DB2 logging to avoid the possibility of any other processes accessing or writing to these disks. Ideal placement of the logs is on the outer edge of the disk where there are more data blocks per track. It is strongly recommended to protect the log against single disk failures by

using RAID 10 or RAID 5 array.

	Description	Example
NEWLOGPATH	This parameter is used to change the log path to create the transaction log files on a separate partition/volume than the default volume or the one used for database tablespace containers. Set it to a directory that is the destination of log files. Make sure that the directory is created before setting it. Make sure that there is enough space on the destination before setting the new log path.	update db cfg for wpcdb using NEWLOGPATH /u02/db2data/logs
LOGFILSIZ	This parameter defines the size of each primary and secondary log file. The size of these log files limits the number of log records that can be written to them before they become full and a new log file is required. Set it to 30000 if it is a development/test database otherwise set it to 60000. The size is number of pages each of size 4 KB.	update db cfg for wpcdb using LOGFILSIZ 30000
LOGPRIMARY	The primary log files establish a fixed amount of storage allocated to the recovery log files. This parameter allows you to specify the number of primary log files to be pre-allocated. Set it to 20 if it is a development database otherwise set it to 40.	update db cfg for wpcdb using LOGPRIMARY 20
LOGSECOND	This parameter specifies the number of secondary log files that are created and used for recovery log files (only as needed). When the primary log files become full, the secondary log files (of size logfilsiz) are allocated one at a time as needed, up to a maximum number as controlled by this parameter. Set	update db cfg for wpcdb using LOGSECOND 10 Bounce the database after making db configuration changes with db2stop and db2start commands

	as controlled by this parameter. Set it to 10 if it is a development/test database otherwise set it to 20.	db2stop force db2start

Install DB2 Admin/Developer/Run-Time Client on Application Server

- Install DB2 Admin/Developer/Run-Time Client on the WebSphere Application Server
- Create any one of the available types of clients

Create a DB2 instance on WAS

Create a db2 instance on the WebSphere Application server. The db2inst1 user or the WebSphere Product Center middleware user can own the instance. You will have to create a 32-bit instance on the app server for the WebSphere Product Center application to connect to the database. You can have a 32-bit instance on the application server connecting to the 64-bit instance on the database server.

Note: If the instance is owned by the db2inst1 user (or any other user other than WebSphere Product Center middleware user) then create a softlink for the WebSphere Product Center middleware user at $HOME/sqllib to the same directory of the instance owner.

Example:

Execute the following command from WebSphere Product Center middleware user home directory

ln -s /home/db2inst1/sqllib/ sqllib

DB2 Database setup check list

Use the following checklist to verify the required DB2 database has been

setup properly for use with WebSphere Product Center.

X	DB2 setup check list	
	Check the DB2 server release	Ensure the DB2 Server release complies with the Installation Requirements outlined in this document
	Check the database codeset	The character set and national character set should be UTF8. Connect as system user and check the character set of the database. (On the Database Server logged in as instance owner) $db2 get db cfg for <database name> This should have "Database code set" set to UTF-8
	Check the parameter file entries	Follow the DB2 configuration sections in this chapter to make sure you have made required parameter changes for the DB2 Registry Variables, Database Manager and the Database.
	Check the tablespaces setup	Make sure the required tablespaces are setup in the database.
	Check the transaction logs setup	Make sure that the transaction logs are created on a separate partition.
	Check the database user setup	View the database user name and password in $TOP/etc/default/common.properties file and make sure that the database user is created and all required privileges are granted to the user.
	Check the connectivity to the database server	The database server and the database server node must be catalogued on the application server and the database must be accessible from the application server. • Check the database connectivity with $TOP/bin/ test_db.sh • Check the JDBC connectivity with $TOP/bin/ test_java_db.sh The database must be accessible from the application server.

Oracle database setup and configuration

OS settings for Oracle

There are several settings for System V semaphores and shared memory that Oracle recommends. These vary by platform and the size of the database. Please consult the Oracle manuals or your DBA for the correct settings.

The following sections define the recommended parameters for the Oracle database operating system:

Oracle on Linux

Edit the following:

/etc/sysctl.conf

Set Parameters:

fs/file-max=16384
kernel/msgmni=1024
kernel/shmmax=3221225472

Note: The value set for kernal/shmmax is recommended if 4GB of memory is available. The size depends on the amount of available memory.

Oracle 9i configuration

This section covers the Oracle database configuration guidelines that are used to properly install WebSphere Product Center.

Satisfy prerequisites

- Make sure the local system satisfies the hardware, software, memory, and disk space requirements for the Oracle Server (Refer to the checklist at the end of this section)
- Install Oracle 9.2.0.5 Enterprise Edition

Use the following guidelines when you create and setup the WebSphere Product Center Database.

Create new database

It is recommended to setup a separate database for the WebSphere Product Center application. One good reason is that WebSphere Product Center Database does not have to depend on the availability and existing configuration (performance tuning point of view) of other databases in use.

Existing Oracle database instances can be used to store WebSphere Product Center data, however due to the length of certain primary keys in the WebSphere schema, the block size must be 8192 KB or larger.

Character set and national character set

WebSphere Product Center uses UTF8 character set. Therefore, the database character set and the national character set must be set to UTF8 at the time of WebSphere Product Center database creation.

WebSphere Product Center specific Oracle parameter file entries (init.ora)

Oracle uses configuration parameters to locate files and specify runtime parameters common to all Oracle products. When an Oracle program or application requires a translation for a particular configuration variable, Oracle consults the associated parameter. All Oracle parameters are stored in the registry.

The following parameters are set for the use with WebSphere Product Center:

- DB_BLOCK_SIZE
- QUERY_REWRITE_ENABLED
- COMPATIBLE
- PROCESSES
- OPEN_CURSORS
- MAX_ENABLED_ROLES
- DB_CACHE_SIZE

- SHARED_POOL_SIZE
- LOG_BUFFER
- SORT_AREA_SIZE
- OPTIMIZER_INDEX_CACHING
- OPTIMIZER_INDEX_COST_ADJ
- OPTIMIZER_FEATURES_ENABLE

	Description	Value
DB_BLOCK_SIZE	The parameter sets the size (in bytes) of an Oracle database block. This value is set at database creation, and cannot be subsequently changed. DB BLOCK SIZE is critical for the Trio schema and must be at least 8192. Schema creation will fail if the db_block_size is too small.	Set it to 8192 for the WebSphere Product Center Database. *Example*: db_block_size = 8192
QUERY_REWRITE_ENABLED	Used to enable or disable query rewriting for materialized views.	This parameter must be set to true. *Example*: query_rewrite_enabled = true
COMPATIBLE	This parameter allows you to use a new release, while at the same time guaranteeing backward compatibility with an earlier release.	Set it to 9.2.0.0.0 or higher. *Example*: Compatible = 9.2.0.0.0
	The parameter specifies the maximum number of operating	Set it to a minimum of 500.

PROCESSES	system user processes that can simultaneously connect to an Oracle Server.	*Example*: Processes = 500
OPEN_CURSORS	The parameter specifies the maximum number of open cursors a session can have at once, and constrains the PL/SQL cursor cache size, which PL/SQL uses to avoid re-parsing statements re-executed by a user.	Set it to 600. *Example*: Open_cursors = 600
MAX_ENABLED_ROLES	Specifies the maximum number of database roles that a user can enable, including sub-roles.	Set it to 60. *Example*: Max_enabled_roles = 60
DB_CACHE_SIZE	This parameter specifies the number of Oracle blocks in the buffer cache. This parameter significantly affects the total SGA size for an instance.	Set it to a value depending upon the total amount of memory available. Set the value at least to 1048576000 *Example*: Db_cache_size = 1048576000
SHARED_POOL_SIZE	The parameter specifies the size of the shared pool in bytes. The shared pool contains objects such as shared cursors, stored procedures, control structures, and Parallel Execution message	Set the value based on the db server memory size. *Example*: Shared_pool_size = 209715200 # 200 MB, if the db server has 2 GB memory

	buffers.	
LOG_BUFFER	Specifies the amount of memory, in bytes, that is used to buffer redo entries before they are written to a redo log file by LGWR. Redo entries keep a record of changes made to database blocks.	Set its value to 5242880. *Example*: Log_buffer = 5242880
SORT_AREA_SIZE	The parameter specifies the maximum amount, in bytes, of memory to use for a sort. After the sort completes, rows are returned and the memory is released. Increase the size to improve the efficiency of large sorts. Temporary disk segments in the users temporary tablespace are used if memory is exceeded.	Set it to a value between 5MB and 10 MB depending upon the main memory available. Setting the sort_area_size too high can cause swapping if too little memory is left over for other processes. *Example*: Sort_area_size = 5242880
OPTIMIZER_INDEX_CACHING	Adjusts the cost-based optimizer's assumptions for what percentage of index blocks are expected to be in the buffer cache for nested loops joins. This affects the cost of executing a nested loops join where	Value: set it to 90 *Example*:

	an index is used. Setting this parameter to a higher value makes nested loops join look less expensive to the optimizer. Range of values is 0 to 100 percent.	Optimizer_index_caching = 90
OPTIMIZER_INDEX_COST_ADJ	Used to tune optimizer performance when too few or too many index access paths are considered. A lower value makes the optimizer more likely to select an index. That is, setting it to 50 percent will make the index access path look half as expensive as normal. Range of Values is 1 to 10000.	Set it to 50 *Example*: optimizer_index_cost_adj=50
OPTIMIZER_FEATURES_ENABLE	Allows init.ora parameters, which control the optimizer's behavior, to be altered.	Set it to 8.1.7 *Example*: optimizer_features_enable=8.1.7

Tablespaces setup

The following tablespaces must be created in the WebSphere Product Center database:

- SYSTEM
- USERS
- INDX
- BLOB_TBL_DATA

- UNDOTBS1
- TEMP

Note: Make sure no data file exceeds size of 1500 MB. Add more data files to the tablespaces to allocate more space.

Tablespace	Description
SYSTEM	This is the default tablespace that is created automatically in the Oracle Database. System tablespace is used to store the data dictionary and the objects created by system user. This is a permanent tablespace. *Recommended: a minimum size of 400MB for the system tablespace.*
USERS	This tablespace is used to store all the WebSphere Product Center Database tables except tables used to store large objects (LOBs). This tablespace is automatically created when you create the database using Oracle Database Configuration Assistant (ODCA). This is a permanent locally managed tablespace. *Recommended: a minimum size of 15 GB for the users tablespace.*
INDX	This tablespace is used to store all the WebSphere Product Center Database indexes. This tablespace is automatically created when you create the database using ODCA. This is a permanent locally managed tablespace. *Recommended: a minimum size of 40 GB for the indx tablespace.*
	This tablespace is used to store WebSphere Product Center Database tables that contain large objects like Catalogs, Images etc. The Oracle Database Configuration assistant does not automatically create this tablespace when you create the database.

BLOB_TBL_DATA	Therefore, make sure to create this tablespace manually after creation of the database. This is a permanent locally managed tablespace. *Recommended: a minimum size of 5 GB for the blob_tbl_data tablespace.*
UNDOTBS1	This tablespace is used to store the rollback segments in an Oracle Database. ODCA automatically creates this tablespace in the database. *Recommended: a minimum size of 15 GB for the undotbs1 tablespace.*
TEMP	This tablespace is used to store objects temporarily in database operations like sorting and grouping. This is also automatically created by ODCA. This is a temporary tablespace. *Recommended: a minimum size of 6 GB for the temp tablespace.*

Oracle tablespace information

Tablespace	Min Size	Recommended storage parameters
SYSTEM	400 MB	Default
USERS	5 GB	EXTENT MANAGEMENT LOCAL SEGMENT SPACE MANAGEMENT AUTO
INDX	20 GB	EXTENT MANAGEMENT LOCAL SEGMENT SPACE MANAGEMENT AUTO
BLOB_TBL_DATA	2 GB	EXTENT MANAGEMENT LOCAL SEGMENT SPACE MANAGEMENT AUTO
		UNDO TABLESPACE

UNDOTBS1	10 GB	LEAVE DEFAULT VALUES
TEMP	5 GB	TEMPORARY TABLESPACE LEAVE DEFAULT VALUES

Redo log files setup

Oracle relies on online redo log files to record transactions. Each time a transaction takes place in the database, an entry is added to the redo log files. Database performance can be increased by correctly tuning the size of the redo log files. Uncommitted transactions also generate the redo log entries. Create six redo log files of size 300 MB each.

Setup the Listener for this database on the database server

WebSphere Product Center connects to the database using JDBC thin client on the app server. Some WebSphere Product Center SQL scripts are also used on the application server for some tasks like creating WebSphere Product Center schema etc. Setup the listener on the DB server so the client can connect to the database using JDBC or SQL Plus.

Create database schema user

Create a database user for WebSphere Product Center, which is referenced in **common.properties** file.

The following is the user information that is needed:

- Default tablespace: users
- Temporary tablespace: temp
- Authentication: Password
- Status: Unlocked
- Roles to be granted: Connect, and Resource
- System Privileges to be granted: unlimited tablespace, select any dictionary and query rewrite

For example execute these SQL commands at the SQL prompt:

```
SQL> Create user WPC identified by WPC default
```

tablespace users temporary tablespace temp;
SQL> Grant connect, resource, unlimited tablespace, select any dictionary, query rewrite to WPC

Install Oracle 9i Client on application server

Install Oracle 9i Client on the application server and make sure you have an entry of the database in tnsnames.ora file on the application server where the Oracle client is installed. The file tnsnames.ora can be found in the "$ORACLE_HOME/network/admin" directory. Check connectivity between the application server and database server using tnsping and/or SQLPlus on the application server.

X	Oracle Setup Check List	
	Check the Oracle 9i server release	The Oracle Server should be Oracle 9.2.0.5 Standard/Enterprise Edition Database Server.
	Check the database character set	The character set and national character set should be UTF8. Connect as system user and check the character set of the database. SQL>select * from nls_database_parameters where PARAMETER in ('NLS_CHARACTERSET,' NLS_NCHAR_CHARACTERSET');
	Check the init parameter file entries	Please go through the Oracle Configuration section in this chapter and make sure the required parameter file entries have been set.
	Check the tablespaces setup	Make sure the required tablespaces are setup in the database.
	Check the rollback segments status	Make sure all the rollback segments are online. Connect as a system user and check the status of the rollback segments. SQL> select SEGMENT_NAME, STATUS from dba_rollback_segs;
		Make sure there is enough redo log files created in

	Check the redo log files	the database. To get information about existing redo log files in the database, connect as a system user and issue the following query: SQL> select * from v$log;
	Check the database user setup	View the database user name and password in $TOP/etc/default/common.properties file and make sure that the database user is created and all required privileges are granted to the user.
	Check the tnsnames.ora file entry for the database	Make sure there is an entry of the database in the tnsnames.ora file on the application server where the Oracle client is installed. The tnsnames.ora file can be found in the following directory: $ORACLE_HOME/network/admin directory. **Note**: Due to a limitation in the schema installation, the service name in tnsnames.ora must match the SID of the database; in other words, OCI utilities such as sqlplus must be able to connect using a service name, which is the same as the SID.
	Check the listener on the database server	The database must be accessible from the application server.

Ch 5 Installing WebSphere Product Center

To install and configure the WebSphere Product Center, perform the tasks as they are listed in this chapter.

Satisfy prerequisites

- Ensure that all installation requirements are fulfilled for your specific operating environment
- Create a user on the WebSphere Application Server with write access to the directory structure of WebSphere Product Center
- It is assumed that bash is used as the default shell script. If using a different shell such as "sh," then execute bash first. Bash must be installed in the directory /bin/bash

Preparing the installation media

Using Passport Advantage

If you obtained your WebSphere Product Center product from Passport Advantage, download the compressed WebSphere Product Center file from Passport Advantage and then uncompress the installation files into the desired installation directory.

General procedures for using the installation media

This section describes general procedures for using the installation media.

Invoking the graphical WebSphere Product Center installer

The graphical WebSphere Product Center Installer presents you with a wizard that allows you to make choices about the installation of WebSphere Product Center. Invoke the Installer by running an executable from one of the installation media installer directories.

Operating System	**Installer directory**
AIX	WPC_AIX
Linux	WPC_Linux
HP-UX	WPC_HP

Example on AIX:

This section describes how to invoke Installer for WebSphere Product Center on an AIX environment.

If you are running CDE on the AIX computer: If you are running the Common Desktop Environment (CDE) and working on the AIX computer directly then you can navigate the installation media and double-click the bin file (*.bin) to invoke the Installer.

You can also navigate the installation media and execute the bin file (.bin) at the command line.

If you are connecting to the AIX computer through X emulation software: If you are using a Windows computer to connect to the AIX computer through X emulation software do the following to invoke the installer:

1. Configure your environment for X emulation software.

2. Execute the bin file (.bin) specific to the operating system. The following example shows how to do so on an AIX computer.

```
./setupAIX.bin
```

The graphical installer starts on the Windows computer that you are using to connect to the AIX computer.

Performing a silent installation

You can perform a silent installation of WebSphere Product Center, where you provide your installation choices in a file rather than at the screens of the installation wizard. This is particularly helpful when you have to perform multiple installations that are identical.

The silent installation can be executed with or without an options file. The options file holds the install parameters used by the silent installation. Run the installer executable at the command line using one of the following choices:

- Set installation options manually
- Use option file to set installation options

Perform silent installation setting options manually

When performing a silent installation, the following options are required:

Installation location	-P installLocation=<dir> **Note**: Use the full path names when specifying <dir> names.
	Include the selection of application server and database combination for a specific operating system. These options can be included as part of the install options set manually or as part of the option file. *For AIX* -W adapterFeatures_aix.selectedFeature=<index> Index values • WebSphere/DB2 = 0 • WebSphere/Oracle= 1 *For Linux*

Application server and database combinations	-W adapterFeatures_linux.selectedFeature=<index> Index values • WebSphere/DB2 = 0 • WebSphere/Oracle= 1 • WebLogic/Oracle= 2 *For HPUX* • WebLogic/Oracle Since there is only one application server and database combination option, it is only reqiured to include the installation location option. ./<installer file> -P installLocation=<dir> -silent **Note**: Refer to the software requirements sections in this document for the versions of each application server and database combination.

To install silently using the required options, use the following command line:

> ./<name of installer file> -W adapterFeatures_<operating system>.selectedFeature=<index> –P installLocation=<dir> -silent

Example

The following example shows how to do so for an AIX computer using the WebSphere/DB2 combination:

> ./setupAIX.bin -W adapterFeatures_aix.selectedFeature=0 –P installLocation=/opt/IBM/WPC –silent

Perform silent install with options file

The silent install can be performed using an options file by using the

following command line:

./<name of installer file> –P -options <options file name >
-silent

The options file must include the required options for the silent install, as discussed in the previous section.

Example

The following example shows how to do so on the AIX computer.

./setupAIX.bin –P –options /home/WPC/install.txt –silent

Record User Selection

The installer can record user selections during an interactive install to a file. This file is generated when installation is complete. This file can be used as a response file for silent installs on other machines (recorded user selection is applied to these installs).

./<name of installer file> –options –record <file name>

Installing WebSphere Product Center

This section provides instructions to install WebSphere Product Center. You can select to install the software product using the graphical installer or silently from the command line.

Installing WebSphere Product Center using the graphical installer

Do the following to install WebSphere Product Center:

1. Invoke the installer as described in "Invoking the graphical WebSphere Product Center installer".

2. Select a language to be used for the installation wizard and click OK.

3. At the "Welcome" screen click **Next**.

4. At the IBM software license acceptance screen, click "I accept the terms in the license agreement" and then click **Next**.

5. Select the application server and database combination from the available list of cobinations and click **Next**.

6. At the product directory screen, perform one of the following tasks:

- Type the full path of the directory into which you want to install WebSphere Product Center in the Directory Name field and click **Next**
- Click Browse to select a directory and click **Next**
- Accept the default path and click **Next**

7. The summary screen lists the features that will be installed, the specified product directory, and the amount of disk space required. Click **Next**.

8. After the Installer finishes successfully, click **Finish**.

Note: Log details are in provided in file "log.txt" located in <install location>/log.txt

---

Set environment variables

Set TOP variable

Purpose: To set the shell variable TOP to point to the WebSphere Product Center installation directory.

Edit the following file:

$HOME/.bash_profile

Note: The name used for the profile above may be different depending on your configuration. It is recommended to use ".bash_profile".

Example on UNIX:

If the WebSphere Product Center is installed in the directory:

/home/user/john/WPCV50

Then, from the command-line:

export TOP=/home/user/john/WPCV50

Source $TOP/setup/init_ccd_vars.sh

Purpose: To define the location of the WebSphere Product Center initialization script.

$TOP/setup/init_ccd_vars.sh

Note: The default shell init script .bash_profile should be modified to set TOP and source init_ccd_vars.sh. Make sure modifications are made to ~/.bash_profile. The location of the file may differ depending on the bash installation.

Edit .bash_profile to include the following:

```
# WPC init

export TOP=<path to WPC{VERSION} directory>

WPC_INIT_VARS=$TOP/setup/init_ccd_vars.sh



if [ -f $WPC_INIT_VARS ]; then

. $WPC_INIT_VARS

else

echo "Error: file '$WPC_INIT_VARS' not found"
```

fi

Edit initialization script (init_ccd_vars.sh)

The "init_ccd_vars.sh file" is sourced to initialize WebSphere Product Center shell variables. For WebSphere Product Center to run properly several key variables, which are listed in this section, must be set in the file "init_ccd_vars.sh". If these variables are not set, errors will occur when starting WebSphere Product Center.

Location of file: $TOP/setup/init_ccd_vars.sh

Set JDK_HOME

Purpose: To define the location of the Java runtime installation. This parameter must be located before the TOP parameter.

Example on UNIX:

export JDK_HOME=/usr/java131

Set Common Unix Utilities

Purpose: To set the properties to the common UNIX utilities used by WebSphere Product Center scripts.

Set the properties for the following utilities:

AWK
CP
ECHO
GREP
MKDIR
RM
SED
TOUCH
TR
BC
CAT

Example on UNIX:

```
export CP=cp
export RM="rm –rf"
export ECHO=echo
export TOUCH=touch
export MKDIR="mkdir –p"
export TR=tr
export GREP=grep
export AWK=awk
export SED=sed
export CAT=cat
export BC=bc
```

Set configuration for use with application server

Purpose: To configure WebSphere Product Center to use WebSphere Application Server

Note: This section only applies to WebSphere Application Server.

Set the following variables:

WAS_HOME	the directory in which WebSphere is installed
WAS_NODENAME	the node name of the WebSphere Application Server. Us local host name
WAS_APPSERVERNAME	the name of the Application Server you are going to use (This must be unique
WAS_VHOST	The name of the WebSphere Virtual Host to which to bir Product Center Middleware. This variable is only needed run multiple WebSphere Product Center middleware serv WebSphere instance. Note: this is not the same as an HTT Host.

Setting memory flags for different modules

All of the different WebSphere Product Center modules are run using a java virtual machine. The user can specify the memory used for these

modules by changing the memory flags provided for this purpose.

For example, the entry,

```
export SCHEDULER_MEMORY_FLAG='-Xmx1024m
-Xms48m'
```

specifies that the SCHEDULER module has the initial memory setting (specified by the option -Xms) of 48MB and the maximum memory setting (specified by the option -Xmx) of 1024MB.

Note: Please refer to the comments in $TOP/setup/init_ccd_vars.sh for more information.

It is recommended to use the following memory flag settings for WebSphere Product Center Services:

```
export ADMIN_MEMORY_FLAG='-Xmx64m -Xms48m'
export APPSVR_MEMORY_FLAG='-Xmx512m -Xms64m'
export EVENTPROCESSOR_MEMORY_FLAG='-Xmx64m
-Xms48m'
export QUEUEMANAGER_MEMORY_FLAG='-Xmx64m
-Xms48m'
export SCHEDULER_MEMORY_FLAG='-Xmx1024m
-Xms48m'
export
WORKFLOWENGINE_MEMORY_FLAG='-Xmx256m
–Xms64m'
```

Configure run time properties

The following steps configure the run time properties used by the system, which are defined in the file common.properties.

Edit the following file:

$TOP/etc/default/common.properties

Setup database information

Purpose: To setup the information allowing access to the database. Tomcat uses this information to login to the database and setup connection pooling.

Set Properties:

db_userName

db_password

db_url

Example:

Assume the username is "wpcapp", password is "track1", the hostname of the database server is "jupiter", and the SID of the Oracle database instance is "pauprd."

The following properties are set:

db_userName=wpcapp

db_password=track1

db_url=jdbc:oracle:thin:@jupiter:1521:pauprd

Note: The referenced port is the listener port. In the example above, the port is 1521.

Set Properties:

db_userName

db_password

db_url

Example:

Assume the username is "wpcapp", password is "track1", the hostname of the database server is "jupiter", and the database instance is "pauprd."

The following properties are set:

db_userName=wpcapp

db_password=track1

db_url=jdbc:db2:pauprd

Setup Timezone

Purpose: To define the time zone property.

Set Property: timezone

Value: set offset in hours or seconds

Example:

To set (GMT-08:00) Pacific Time (US & Canada); Tijuana in hours:

timezone=-8

To set (GMT-08:00) Pacific Time (US & Canada); Tijuana in minutes:

timezone=-480

Setup FTP Directory

Purpose: To define the directory where data files are transferred to using ftp (for an import).). If FTP is not used for data import, this property can be left at the default. If FTP is not used for data import, this property can be left at the default.

Set property:

ftp_root_dir

Note: Files are stored in

/u01/ftp/COMPANY_CODE/files will show up in the ftp section of the Docstore. If files are put into /u01/ftp, they will not show up. Therefore, the property must be set to the organizations ftp directory.

For example, /u01/ftp/wpc/files.

Example on UNIX:

ftp_root_dir=/u01/ftp/

Note: The ftp directory ends with a forward slash '/'.

WebSphere Product Center reads from and writes to this FTP directory.

Set this directory to allow read/write privilege for WebSphere Product Center users.

Setup Tmp Directory

Purpose: This setting identifies the temporary directory on the server.server.

Set property:

tmp_dir

Example on UNIX:

tmp_dir=/tmp/

Note: The tmp directory ends with a forward slash '/'.

Set this directory to allow read/write privilege for WebSphere Product Center users.

Test for valid WebSphere Product Center license file

The script $TOP/bin/go/start_local.sh script is used to start all of the WebSphere Product Center middleware services (or modules) and is required to run WebSphere Product Center. A WebSphere Product Center module can be successfully started only if there exists a valid license file $TOP/etc/default/ccd_license.xml that specifies that the module is valid. If it is invalid, the module will fail to start. The errors are logged in the specific log file for each module in the directory: $TOP/logs/svc.

Note: License files are provided separately from the installation files. Contact WebSphere Product Center support if a license file is needed to operate WebSphere Product Center.

To test the validity of the WebSphere Product Center License file, run the script:

$TOP/bin/test_license.sh

If the IP is not correct, the license validation reports the expected IP and the current local IP.

Application server configuration

This section provides instructions required to run WebSphere Product Center with the application server.

IBM WebSphere Application Server Configuration

In order to use WebSphere Product Center with WebSphere Application Server, follow the configuration instructions described in the following sections.

- Verify configuration settings
- Start WebSphere default server and Administrative Console
- Add WebSphere group and change WebSphere settings

- Run WebSphere Scripts
- Start WebSphere application

Verify configuration settings

In addition to the standard variables, verify that the following configuration settings are defined in $TOP/setup/init_ccd_vars.sh:

Name	Value
WAS_HOME	home directory of WebSphere. I.E. /usr/WebSphere/AppServer
WAS_APPSERVERNAME	name of the Application Server you are going to use (within websphere). This must be unique.
WAS_NODENAME	The node name of the WebSphere Application Server on which you are running. Usually this will be the same as the system's hostname
WAS_VHOST	The name of the WebSphere Virtual Host to which to bind the WebSphere Product Center Middleware. This variable is only needed if you are going to run multiple WebSphere Product Center middleware servers on the same WebSphere instance. Note: this is not the same as an HTTP server Virtual Host.

Start default server and administrative console

To start the WAS default server, issue the following command as root:

```
${WAS_HOME}/bin/startServer.sh server1
```

Add WAS group

On AIX, you can add a group using the SMIT administration tool.

Note: Make sure the CCD user is part of the WAS group.

- Log into the WebSphere administrative console. (normally runs at http://your-server-name:9090/admin)
- Click Servers -> Application Servers
- Click on the default server (server1)
- Click "Process Definition"
- Click "Process Execution"
- Change UMASK to "002".
- Change "Run as Group" to the WAS group you created. If the WebSphere default server is not running as root, the WebSphere user should also be in the WAS group.
- Change the permissions on the WebSphere directory so that the WAS group has write permission.

  # chgrp –R wasgroup ${WAS_HOME}

  # chmod –R g+w ${WAS_HOME}

Run WAS scripts

After completing the previous WebSphere Product Center installation steps, run the following scripts in the listed order:

1. $TOP/bin/websphere/create_appsvr.sh
2. $TOP/bin/websphere/create_vhost.sh
3. $TOP/bin/websphere/install_war.sh

Running multiple instances of WebSphere Product Center under the same WebSphere server

To run multiple instances of WebSphere Product Center using one WebSphere server, the following tasks must be performed:

- **Requires that the WebSphere Administration Server be running as root.**
- Install each instance normally as above, but do not yet run create_appsvr.sh or install_war.sh
- Make sure each instance has a unique WAS_APPSERVERNAME and WAS_VHOST defined in init_ccd_vars.sh, and unique application ports defined in common.properties
- Run the following scripts in the following order under each

instance:

1. $TOP/bin/websphere/create_vhost.sh
2. $TOP/bin/websphere/create_appsvr.sh
3. $TOP/bin/websphere/install_war.sh

- Create a suppliers symlink for each instance. (if install_war.sh fails to create symlink)

BEA WebLogic Configuration

This section describes the required tasks to properly configured WebLogic for a Websphere Product Center environment.

- Set default home directories
- Create a new domain with the $WL_HOME/common/bin/dmwiz.sh wizard
- Update the **config.xml** file
- Copy or link the WebLogic domain applications directory
- Verify public files directory

For the purpose of this document, the WebLogic setup instructions in this section are configured for the following:

- Domain name 'wpc_domain'
- Listening on port '7507'
- Domain files contained in the directory '/wpc/envs/wl_username/wl_domain'
- Servername set to wpc_server'
- WebLogic username 'wl_username'
- WebLogic password 'wl_password'

Set default home directories

To set the default home directories for WebLogic , configure the following parameters in $TOP/setup/init_ccd.vars.sh:

- WPC_WL_DOMAIN_DIR (*home of the WebLogic WPC domain*)
- WPC _WL_DOMAIN (*domain name for the WPC domain*)
- WPC _WL_SERVER (*server name for the WPC domain*)

- WPC _WL_PW (*password to start the WebLogic server*)
- WPC _WL_USER (*WebLogic server user name*)

Example:

```
export
WPC_WL_DOMAIN_DIR="/wpc/envs/wl_username/wl_domain/wpc_domain"
export WPC_WL_DOMAIN=wpc_domain
export WPC_WL_SERVER=wpc_server
export WPC_WL_PW=wl_password
export WPC_WL_USER=wl_username
```

Create a new WebLogic domain

To create a new WebLogic domain, use the dmwiz.sh wizard located in the following directory:

$WL_HOME/common/bin/dmwiz.sh

1. Run $WL_HOME/common/bin/dmwiz.sh

2. **Choose Domain Type** (3) WLS Domain

3. **Choose Server Type** (1) Single Server (Standalone Server)

4. **Choose Domain Name** (1) Modify current selection

5. **Configure Single Server**

6. **Modify Server Name** wpc_server

7. **Modify Server Listen Port** 7507

8. **Choose Location to Create Domain** > **Modify current selection** /wpc/envs/wl_username/wl_domain

9. **Enter User Name** wl_username

10. **Enter Password** wl_password

11. **Enter Verify Password** wl_password

12. Press Enter to create the domain and follow the prompts to exit the wizard.

At the completion of the dmwiz wizard, the following message is displayed: **Domain Configuration Wizard has successfully created "wpc_domain"**

Now that the WebLogic domain has been created in "/wpc/envs/wl_username/wl_domain/wpc_domain", it can be configured by editing " /wpc/envs/wl_username/wl_domain/wpc_domain/config.xml"

Configure config.xml

1. Remove the following sections

- SSL section
- Application DefaultWebApp section
- Certificate application section

2. Add the following "Application" and "Startup Class" sections:

- Applications (add within the Domain xml parameters)

  ```
  <Application Deployed="true" Name="suppliers"

  Path="{ACTUAL_PATH_TO_PUBLIC_HTML}" TwoPhase="false">

  <WebAppComponent Name="suppliers" Targets="wpc_server"
  URI="suppliers"/>

  </Application>

  <Application Deployed="true" Name="wpc" Path="./applications"

  StagedTargets="" TwoPhase="false">

  <WebAppComponent Name="wpc" Targets="wpc_server"
  URI="wpc.war"/>

  </Application>
  ```

- Startup Class (add within the Domain xml parameters)

```
<StartupClass
ClassName="austin.appsvr.current.admin.RemoteWeblogic"

FailureIsFatal="true" Name="RemoteWeblogic"
Targets="wpc_server"/>
```

3. Change DefaultWebApp

Change the DefaultWebApp in the WebServer field from "DefaultWebApp" to "wpc"

4. WebLogic logging -If desired, change the log parameter to redirect the weblogic.log file to a different location.

Sample config.xml file after configuration

```
<!--If your domain is active, please do not edit the config.xml file. Any changes
made to that file while the domain is active will not have any effect on the
domain's configuration and are likely to be lost. If your domain is inactive, you
may edit this file with an XML editor. If you do so, please refer to the BEA  Server
Configuration Reference documentation available from
http://e-docs.bea.com/wls/docs61/config_xml/index.html. In general, we
recommend that changes to your configuration file be made through the
Administration Console.-->

<Domain Name="wpc_domain">

 <Server
  Name="wpc_server"
  NativeIOEnabled="true"
  ListenAddress=""
  ListenPort="7507" >

   <WebServer
   DefaultWebApp="wpc"
   LogFileName="access.log"
   LoggingEnabled="true"
   Name="wpc_server"
   />

  <Log
  FileName="weblogic.log"
  />
  </Server>

  <Application Deployed="true" Name="suppliers"
```

```
 Path="/wpc/envs/wl_username/phtml" TwoPhase="false">
 <WebAppComponent Name="suppliers" Targets="wpc_server"
URI="suppliers"/>
 </Application>
 <Application Deployed="true" Name="wpc" Path="./applications"
 StagedTargets="" TwoPhase="false">
 <WebAppComponent Name="wpc" Targets="wpc_server" URI="wpc.war"/>
 </Application>

 <Security
 Name="wpc_domain"
 GuestDisabled="false"
/>

 <Log
 FileName="wl-domain.log"
 />

 <StartupClass ClassName="austin.appsvr.current.admin.RemoteWeblogic"
 FailureIsFatal="true" Name="RemoteWeblogic" Targets="wpc_server"/>

</Domain>
```

WebLogic Domain applications directory

Copy the WebLogic Domain applications directory ($TOP/etc/default/weblogic700/config/wpc_domain/applications/ directory) into the new domain directory.

To create a symbolic link to the applications directory, delete the applications directory and replace with a symbolic link to the WebSphere Product Center code base.

For Example (Using the example in the previous section - (/wpc/envs/wl_username/wl_domain/wpc_domain/applications):

```
wl_username@app02:~/wl_domain/wpc_domain$ mv applications
applications.bk/

wl_username@app02:~/wl_domain/wpc_domain$ ln -s
/wpc/envs/wl_username/code/wpc42010009/etc/default/weblogic700/config/wpc_domain/applica
.

wl_username@app02:~/wl_domain/wpc_domain/applications$ ls -l
```

```
total 28896

-rw-r--r-- 1 wl_username envs 29553587 Dec 31 13:36 wpc.war

wl_username@app02:~/wl_domain/wpc_domain/applications$
```

Verify public files directory

Verify that the directory defined as the 'suppliers' application is the same as the directory used by WebSphere Product Center for public_html. For example, create a symbolic link from $TOP/public_html into ~/phtml

```
wl_username@app02:~/code/wpc42010009$ mv public_html/ public_html.bk

wl_username@app02:~/code/wpc42010009$ ln -s ~/phtml/ public_html

wl_username@app02:~/code/wpc42010009$
```

WebLogic Issues

WebSphere Product Center disappears

If the applications directory is not located in the proper place, WebLogic automatically removes the WebSphere Product Center. The solution is to re-add the WebSphere Product Center in the config.xml file, and to re-set the default web application to WebSphere Product Center.

Embedded LDAP lock error

Startup gives the following error -

(...)ldap/ldapfiles/EmbeddedLDAP.lok, No locks available,errno=37

A solution for this a locking issue is to move the wpc_server directory to a local disk. This can be sym linked from the NFS wpc_domain.

Setup database schema

Note: The file common.properties must be setup before creating the

database schema. (Refer to the section "Configure Run Time Properties" in this chapter)

Verify database user

Verify that the database user referenced in common.properties exists in the database with the proper privileges.

Execute the following shell script to test that WebSphere Product Center can talk to the database:

$TOP/bin/test_java_db.sh

The script attempts to parse common.properties and login to your database.

- If the script produces an error, please modify the common.properties file to set the correct database user, or verify with you Database Administrator that the database user was created correctly.

Create database schema

Once the DB user is setup correctly, the WebSphere Product Center database schema can be created.

Execute the shell script:

$TOP/src/db/schema/create/create_schema.sh

The script first shows which database it is trying to connect to. Then the system prompts the user to confirm with "y" to continue or "n" to stop.

Type "y" to continue.

Note: Once the database schema is created, there is no need to run create_schema.sh again. The database schema only needs to be created once.

Setup a company to test

This section sets up a company that is used to test the WebSphere Product Center installation. A company must be created in order to login to WebSphere Product Center. To create a new company, refer to the section "Create a new company".

Create a company to test

To login to WebSphere Product Center, a company must be created. Do one (or both) of the following to create a company:

Create Acme Company

To create a demo company called "acme", execute the following shell script:

$TOP/src/db/schema/acme/create_acme.sh --code=acme

The company Acme is created with four users:

Admin, bwilson. Jwilson, vjackson

All users are provided the password "trinitron"

Create Empty Company

To create an empty company called "demo", with no predefined data (quicker than creating the demo acme company), execute the following shell script:

$TOP/src/db/schema/cmp/create_cmp.sh --code=demo

The company demo is created with a single user:

Admin

The password for Admin is "trinitron"

Note: Passwords are case-sensitive. The Admin user is

created with full privileges and should be used by an administrator.

Test WebSphere Product Center Installation

Start WebSphere Product Center

To start WebSphere Product Center, execute the script start_local.sh, which starts all the services needed to run WebSphere Product Center.

$TOP/bin/go/start_local.sh

Verify that the WebSphere Product Center is up and running by executing the following script:

$TOP/bin/go/rmi_status.sh

Verify that the following services have started:

- admin
- appsvr
- eventprocessor
- queuemanager
- scheduler
- workflow

An example of normal output from start_local.sh:

```
#./start_local.sh
killing services on localhost
killing service 'appsvr_LORAX'
killing service 'admin_LORAX'
killing service 'ajp12_LORAX'
killing service 'ajp13_LORAX'
killing service 'eventprocessor'
killing service 'queuemanager'
killing service 'scheduler'
killing service 'rmi'
```

```
starting rmiregistry on port 17507
starting service 'admin_LORAX'
starting service 'appsvr_LORAX'
starting service 'eventprocessor'
starting service 'queuemanager'
starting service 'scheduler'
```

Verify that the application is up and running by executing the following script:

```
/usr/trigo/pink_tulip/austin/bin/go/rmi_status.sh
```

Verify that the following services have started:

```
admin_LORAX
appsvr_LORAX
eventprocessor
queuemanager
scheduler
```

Note: This process should take approximately 30-40 seconds, depending on the speed of the processor.

Check RMI status

Verify that the WebSphere Product Center is up and running by executing the following script:

```
$TOP/bin/go/rmi_status.sh
```

This script contacts the RMI daemon on all the machines in the cluster and obtains a list of the local services on each machine. An extended list of names is returned.

The following is an example of rmi_status.sh showing that all services have started on a machine named "LORAX":

```
#./rmi_status.sh
++ [success] rmistatus (Mon Aug 26 17:29:47 PDT 2003)
rmi://lorax:17507/CMP1/appsvr/appsvr_LORAX
```

rmi://lorax:17507/CMP1/admin/admin_LORAX
rmi://lorax:17507/CMP1/eventprocessor/eventprocessor_LORAX
rmi://lorax:17507/CMP1/scheduler/scheduler_LORAX
rmi://lorax:17507/CMP1/queuemanager/queuemanager_LORAX
rmi://lorax:17507/CMP1/workflow/workflow_LORAX

Test database connection

To test the database connection using Java to validate the driver, run the following script.

```
$TOP/bin/test_java_db.sh
```

Login to WebSphere Product Center

Open the Internet Explorer web browser and enter the URL and port for the web server.

```
http://www.acme.com:7507
```

Note: During the WebSphere Product Center installation, the web server port was set to 7507 in a two tier configuration. If a different port is used, change the port reference in the file server.xml or in the Apache configuration file for a three tier configuration

Using the Acme company demo, login as:

Username: bwilson
Password: trinitron
Company Code: acme

If the login to the WebSphere Product Center opens up the WebSphere Product Center Home Page, the installation was successful. Logout of the application and continue to the next step.

At this point, if all connections are running properly, open the Internet Explorer web browser and enter the URL and port for the WebSphere Product Center server.

```
http://www.acme.com:7507
```

If the create_acme.sh was used to create the Acme company demo, login as:

Username: bwilson
Password: trinitron
Company: acme

If the create_cmp.sh was used to create a company, login as:

Username: Admin
Password: trinitron
Company: acme

If the login to the WebSphere Product Center opens up the WebSphere Product Center Home Page, the installation was successful.

Stop the WebSphere Product Center

To stop WebSphere Product Center on the local machine, execute the following script:

$TOP/bin/go/stop_local.sh

The script stops all the services started by WebSphere Product Center. Example output from stop_local.sh showing all services stopped:

```
#./stop_local.sh
stopping services on localhost
++ [success] stop service 'appsvr_LORAX' (Mon Aug 26 17:55:46 PDT
2002)
Websphere will stop in 5 seconds
++ [success] stop service 'admin_LORAX' (Mon Aug 26 17:55:47 PDT
2002)
admin will stop in 5 seconds
++ [success] stop service 'eventprocessor' (Mon Aug 26 17:55:47 PDT
2002)
event processor stopped
++ [success] stop service 'scheduler' (Mon Aug 26 17:55:47 PDT 2002)
scheduler will stop in 5 seconds
++ [success] stop service 'queuemanager' (Mon Aug 26 17:55:48 PDT
```

2002)
queue manager stopped
killing service 'rmi'

Create a new company

Now that WebSphere has been installed and tested, create a new company for use with a test or production environment. If desired, more than one company can be created.

For example, to create an empty company called "test", with no predefined data, execute the following shell script:

```
$TOP/src/db/schema/cmp/create_cmp.sh --code=test
```

The company is created with a single default administrator user:

Username: Admin
Password: trinitron

Note: The Admin user is automatically created with full privileges. This user account is to be used by an administrator.

Ch 6 - Uninstalling WebSphere Product Center

This chapter describes how to uninstall WebSphere Product Center. The uninstaller can run using the graphical WebSphere Product Center Uninstaller or silently from the command line.

Invoking the graphical uninstaller

The graphical WebSphere Product Center Uninstaller presents you with a wizard that allows you to uninstall WebSphere Product Center. The Uninstaller is invoked by running executables from the WebSphere Product Center installation directory.

Example in UNIX

This section describes how to invoke Uninstaller for WebSphere Product Center on a Unix environment.

If you are running CDE on the UNIX computer: If you are running the Common Desktop Environment (CDE) and working on the UNIX computer directly then you can navigate to the uninstallation directory of a WebSphere Product Center installation and double-click the .bin file to invoke the Uninstaller.

If you are connecting to the UNIX computer through X emulation software: If you are using a Windows computer to connect to the UNIX computer through X emulation software do the following to invoke the installer:

1. Configure your environment for X emulation software.

2. Execute <install location>/_uninst/uninstaller.bin

The graphical uninstaller starts on the Windows computer that you are

using to connect to the UNIX computer.

Invoking the graphical uninstaller at the command line

To execute the graphical uninstaller for WebSphere Product Center, execute the following command at the command line:

<install location>/_uninst/uninstaller.bin

Using the graphical uninstaller

Do the following to uninstall WebSphere Product Center using the graphical uninstaller:

1. At the "Welcome" screen click **Next**.
2. The "Summary " screen lists the components that will be uninstalled and the product directory from which they will be removed. Verify the information and then click **Next**.
3. After the uninstaller finishes successfully, click Finish.

Performing a silent uninstallation

To perform a silent uninstallation for WebSphere Product Center, execute the following command at the command line:

<install location>/_uninst/uninstaller.bin

Ch 7 Administrating WebSphere Product Center services

Service types

The complete WebSphere Product Center system consists of the following services running concurrently:

admin	the admin server starts/stops modules on remote machines
appsvr	the application server serves Java Server Pages
eventprocessor	the event processor dispatches events between all the modules
queuemanager	the queue manager sends documents outside of WebSphere Product Center
scheduler	the scheduler runs background jobs
workflow	the workflow engine

admin_properties.xml and clustering

Services can run in a cluster of workstations. The different machines in the cluster are defined in the file admin_properties.xml:

$TOP/etc/default/admin_properties.xml

Note: Additional information is provided within admin_properties.xml Each service can run on any of the machines listed in the admin_properties.xml file.

A typical WebSphere Product Center cluster can contain the application server and the support RMI Registry utility on the WebSphere Product Center server, and the remaining of the WebSphere Product Center

components on the secondary server.

In the case of a primary server failover, the services that were not previously running on the secondary server could be brought back online with minimal effort, minimizing downtime.

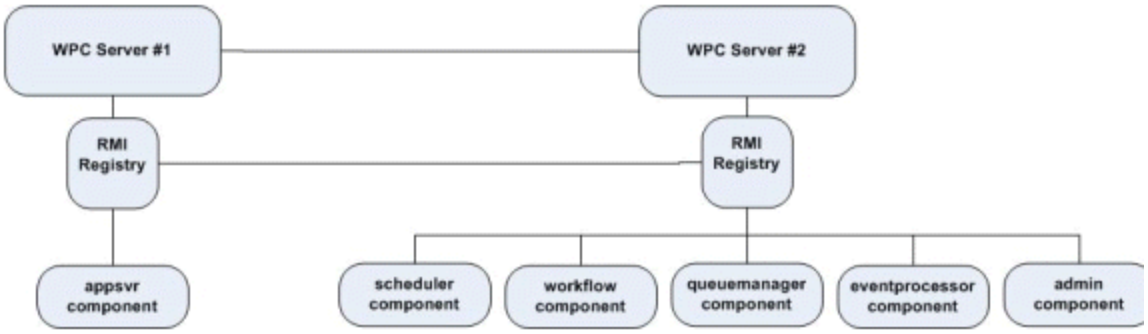

Figure 4 - Typical WebSphere Product Center cluster

Service name - long and short name

Each service is uniquely identified by a service name. The service name must be unique (a service will refuse to start if another service with the same name is running on a machine in the cluster.)

Each service can run on multiple machines as long as the service name is different.

The name for the 'admin' and the 'appsvr' services are fixed by the system.

admin_<machine name> for admin (ex: 'admin_server1')

appsvr_<machine name> for appsvr (ex: 'appsvr_server1')

For other services, pick an arbitrary name. The name selected is actually the service's short name.

Internally, a long name is built using this short name:

rmi://<machine name>:<rmi port>/<db user name>/<service

type>/<service short name>

Example:

If you are running a 'scheduler' service on a machine 'server1', using the rmi port 17507, connecting to a database user 'pauadm', and name the service 'sch1', the long name is:

rmi://server1:17507/pauadm/scheduler/sch1

If another scheduler (sch2) runs on server 2 for the same user and port, the long name is:

rmi://server2:17507/pauadm/scheduler/sch2

Setting memory flags for service types

Memory flags for various WebSphere Product Center services are set in the WebSphere Product Center initialization script located in the WebSphere Product Center installation directory.

<install location>/setup/init_ccd_vars.sh

It is recommended to use the following memory flag settings for WebSphere Product Center Services:

```
export ADMIN_MEMORY_FLAG='-Xmx64m -Xms48m'

export APPSVR_MEMORY_FLAG='-Xmx512m -Xms64m'

export EVENTPROCESSOR_MEMORY_FLAG='-Xmx64m
-Xms48m'

export QUEUEMANAGER_MEMORY_FLAG='-Xmx64m
-Xms48m'

export SCHEDULER_MEMORY_FLAG='-Xmx1024m
-Xms48m'

export
WORKFLOWENGINE_MEMORY_FLAG='-Xmx64m
```

-Xms48m'

RMI – Remote Method Invocation

Service registration is done through RMI (Java Remote Method Invocation). Before running any services, make sure that RMI is started on the machine.

RMI status

To get a list of all the running services in a cluster, execute the following script:

```
$TOP/bin/go/rmi_status.sh
```

This script contacts the RMI daemon on all the machines in the cluster and gets a list of the local services on each machine. It returns a list of long names.

Log files

Each service will produce a run time log file

```
$TOP/logs/<service>/<service
name>/svc.out
```

Example:

A scheduler named 'sch1' produces a runtime log file svc.out in $TOP/logs/scheduler/sch1

After starting a service, it is recommended to check the log file to make sure that everything started without any problems.

Starting a service

The following sections describe how services are controlled using local scripts. Before a service can be used, the RMI registry has to be started on the machine using the service.

To start RMI, run the script:

```
$TOP/bin/go/start/start_rmiregistry.sh
```

Starting a service on the local machine

The simplest way to start a service on the local machine is to use the scripts in the directory $TOP/bin/go/start/

Script	Description
start_admin.sh	starts admin service
start_appsvr.sh	starts application server
start_eventprocessor.sh	starts event processor
start_queuemanager.sh	starts queue manager
start_rmiregistry.sh	starts RMI registry
start_scheduler.sh	starts scheduler
start_workflowengine.sh	starts workflow engine

Each of these scripts (except start_admin.sh, start_appsvr.sh, and start_rmiregistry.sh) can take the service name as an optional argument:

-svc_name=<service name>

The admin and appsvr services use a default name (admin_<machine name> and appsvr_<machine name>). Specifying another name will have no effect.

If no service name is specified, it uses a default name:

"scheduler" for the scheduler

"eventprocessor" for the event processor

"queuemanager" for the queue manager

"workflow" for the workflow engine

Note If a local service is started, with the name of a local service that is already running, the previous local service will be aborted first. Therefore the scripts can be used to 'restart' a service as well (abort first,

and then restart)

Example:

To start the scheduler with the name "sch1":

```
$TOP/bin/go/start/start_scheduler.sh
-svc_name=sch1
```

To start the scheduler with the default name:

```
$TOP/bin/go/start/start_scheduler.sh
```

Aborting a service

Aborting a service shuts down the service and becomes unavailable.

For example, if the scheduler is running a job, the job will be aborted in the middle of the process.

Aborting a service on the local machine

The structure here mirrors that start structure.

Use the scripts in the directory $TOP/bin/go/abort/

Script	**Description**
abort_admin.sh	aborts admin service
abort_appsvr.sh	aborts application server
abort_eventprocessor.sh	aborts event processor
abort_queuemanager.sh	aborts queue manager
abort_rmiregistry.sh	aborts RMI registry
abort_scheduler.sh	aborts scheduler
abort_workflowengine.sh	aborts workflow engine

Each of these scripts (except abort_admin.sh , abort_appsvr.sh, and abort_rmiregistry.sh) can take the service name as an optional argument:

-svc_name=<service name>

Note: Aborting RMI makes it impossible to contact services on remote machines.

Stopping a service

Stopping a service will request the service to shut down smoothly. If the service is "blocked", it might not execute the shut down procedure at all. The scheduler will not stop until it has finished executing all the current running jobs.

Stopping a service on the local machine

The structure here mirrors the start structure.

Use the scripts in the directory $TOP/bin/go/stop/

Script	**Description**
stop_admin.sh	stops admin service
stop_appsvr.sh	stops application server
stop_eventprocessor.sh	stops event processor
stop_queuemanager.sh	stops queue manager
stop_scheduler.sh	stops scheduler
stop_workflowengine.sh	stops workflow engine

Each of these scripts (except abort_admin.sh , abort_appsvr.sh, and abort_rmiregistry.sh) can take the service name as an optional argument:

-svc_name=<service name>

Important note on abort and stop

Which one should be used? Stop or abort?

abort	Guarantees that the service will be shut down, but it cannot guarantee that a currently running task will not be interrupted.

stop	Guarantees that "if" the service is stopped, it will be stopped smoothly after every currently running task has been stopped first.

Starting the all WebSphere Product Center modules

Starting the WebSphere Product Center on the local machine

Run the script `$TOP/bin/go/start/start_local.sh`

This will start the RMI registry as well as the following services:

- admin named 'admin_<machine name>'
- application server named 'appsvr_<machine name>'
- event processor named 'eventprocessor'
- queue manager named 'queuemanager'
- scheduler named 'scheduler'
- workflow named 'workflow'

Note: It will first try to kill any existing system on the local machine before starting anything.

Aborting WebSphere Product Center on the local machine

Run the script `$TOP/bin/go/abort/abort_local.sh`

Every service started on the local machine is aborted. The RMI registry is aborted.

Stopping WebSphere Product Center on the local machine

Run the script `$TOP/bin/go/stop/stop_local.sh`

Every service started on the local machine is stopped. By default, the RMI registry is stopped with the other services. To keep the RMI registry running, pass the following option:

--kill_rmi=no

Note: There are two dashes before "kill_rmi=no"

Service status

Getting the short status of a service

To get the short status of a service, pass the following parameters:

-cmd=check -svc=<service name>

Example:

To get the status of the scheduler:

rootadmin.sh -cmd=check -svc=scheduler

The short status can be one of the following:

running	The service is running and responding to a "heartbeat" function.
not found	The service is not found. The service might not have been started or it might have crashed.
found but not responding	The service was found as being registered with the RMI registry, but it is not responding to the "heartbeat" function. The service might have to be restarted.

Getting the long status of a service

To get the long status of a service, pass the following parameters:

-cmd=status -svc=<service name>

It will produce an html file that can be viewed using any browser. On a terminal, you might want to use lynx to format the output.

Example:

To get the status of the scheduler:

rootadmin.sh -cmd=status -svc=scheduler > /tmp/sch_status.html; lynx /tmp/sch_status.html

OR

```
rootadmin.sh -cmd=status -svc=scheduler >
/tmp/sch_status.html; lynx -dump /tmp/sch_status.html
```

Note: The ">" used in the example above directs the status details to a file output location.

The status gives you an overview of the different threads running in the service, as well as a status of the database connections currently taken by the service.

Appendix A - Configuration reference

This section outlines the parameters that are used in various WebSphere Product Center property files.

- common.properties
- admin_properties.xml
- docstore_mount.xml
- data_entry_properties.xml

Many of the values are assigned to parameters during the installation of WebSphere Product Center. If any values need to be modified, they can be made using the rules outlined in this Appendix.

File Name: common.properties

On startup, the system will use this file to read in all system level parameters. This file includes settings for the database layer (connection parameters), directory settings, default character sets, thread-pooling parameters, and other settings.

Note: The common.properties parameters are listed in the order as they appear in the file.

Database setup

Purpose: Connect to database

db_username - The user name used to login to the database

db_password - The password used to login to the database

db_url jdbc url - used to connect to the database. Use the following format: jdbc:db2:<db alias>

Example:

db_userName=qa1

db_password=qa1

db_url=jdbc:db2:qadb

Separate tablespace

Purpose: To define if the database has different tablespaces. This parameter is useful for laptop or stand-alone systems

f false, then GenSchema.java ignores all tablespace directives

If true, then GenSchema.java observes all tablespace directives

Example:

db_tablespaces=true

DB parameters for default service

Purpose: Defines the parameters for a default service

Value: Use an integer value

Example:

db_maxConnection=8

db_minConnection=2

db_maxConnection_default=4

db_maxConnection_system=4

DB parameters for admin service

Purpose: Defines the parameters for a admin service

Value: Use an integer value

Example:

db_maxConnection_admin=5

db_minConnection_admin =2

db_maxConnection__admin_default=4

db_maxConnection__admin_system=1

DB parameters for app server service

Purpose: Defines the parameters for an app server service

Value: Use an integer value

Example:

db_maxConnection_appsvr=30

db_minConnection_appsvr=10

db_maxConnection_appsvr_default=26

db_maxConnection_appsvr_system=4

DB parameters for scheduler service

Purpose: Defines the parameters for a scheduler service

Value: Use an integer value

Example:

db_maxConnection_scheduler=40

db_minConnection_scheduler=10

db_maxConnection_scheduler_default=36

db_maxConnection_scheduler_system=4

DB parameters for eventprocessor service

Purpose: Defines the parameters for an eventprocessor service

Value: Use an integer value

Example:

db_maxConnection_eventprocessor=6

db_minConnection_eventprocessor=4

db_maxConnection_eventprocessor_default=2

db_maxConnection_eventprocessor_system=4

DB parameters for queue manager service

Purpose: Defines the parameters for the queue manager service.

Value: Use an integer value

Example:

db_maxConnection_queuemanager=12

db_minConnection_queuemanager=4

db_maxConnection_queuemanager_default=6

db_maxConnection_queuemanager_system=6

DB parameters for workflow engine service

Purpose: Defines the parameters for the workflow engine service.

Value: Use an integer value

Example:

db_maxConnection_workflowengine=12

db_minConnection_workflowengine=4

db_maxConnection_workflowengine_default=6

db_maxConnection_workflowengine_system=6

DB connection pool info

Purpose: Defines the parameters for the DB connection pool info.

Value: The max connection time is in days.

Example:

db_maxConnTime=1

db_numRetry=3

db_retrySleep=

db_class_name=COM.company.db2.jdbc.app.DB2Driver

Set time zone

Purpose: This is only read if the user has not specified a default user-setting time zone.

If this value is not specified, then the code is hard coded to default to PACIFIC which is –8

Use an integer relative to GMT (Greenwich Mean Time)

-8 for PACIFIC

-7 for MOUNTAIN

-6 for CENTRAL

-5 for EASTERN

- 2 for ENGLAND

- 1 for CENTRAL EUROPE

Example:

timezone=-8

Number of rows per page

Purpose: The number of rows per page in different tables. These values are read if there is no value specified in the user setting for that user. If these are also not specified, then a hardcoded value of 20 is used.

Values: Enter an integer

Example:

table_rows_per_page_specs_console=20

table_rows_per_page_scripts_console=20

table_rows_per_page_item_set=20

table_rows_per_page_multi_edit=10

table_rows_per_page_alerts_display=25

table_rows_per_page_lookup_table=20

Flush performance buffer

Purpose: To define how long before the performance buffer is flushed.

Value: Units are number of writes to the buffer, i.e. every 100 writes then flush.

Example:

db_perf_dump=100

WebSphere Product Center log files

Purpose: To define the location for WebSphere Product Center log files, relative to ${TOP}.

Values: Enter the directory for the WebSphere Product Center log file

Example:

dbpool_log_file=logs/db_pool/pool_log.txt

Base directory for each supplier (relative to ${TOP})

Purpose: To define the base directory for each supplier, relative to ${TOP}.

> Note: The based directory must start with public_html.

Value: Enter the base directory path for each supplier, relative to ${TOP}

Example:

supplier_base_dir=/public_html/suppliers/

FTP root directory

Purpose: To define the root directory for FTP

Value: The path of the FTP root directory must end with a forward slash"/"

Example:

ftp_root_dir=/u01/ftp/

Scheduler info

Purpose: Num_threads is the max number of scheduler worker threads, master_poll_time and scheduler_nap_log_factor are used to tune how long the scheduler waits after starting a job and before looking for another schedule to run. The master_poll_time is also used as the wait between successive queries to the DB.

The idea is that we want to avoid having a single scheduler start 10 jobs when another scheduler has only 1 job

nap_time = master_poll_time + scheduler_nap_log_factor*log(count_busy_threads)

Example:

Log is the log base 10

With 1 busy threads, log(1) = 0, and nap_time = 5000 + 10000*0 = 5 seconds

With 10 busy threads, log(10) = 1, and nap_time = 5000 + 10000*1 = 15 seconds

With 100 busy threads, log(100) = 2, and nap_time = 5000 + 10000*2 = 25 seconds

Parameters:

master_poll_time

sch_poll_time

num_threads

Values: Enter an integer value. The times are in millisecond.

Example:

master_poll_time=5000

scheduler_nap_log_factor=10000

sch_poll_time=30000

num_threads=8

Workflow engine worker thereads

Purpose: To define the maximum number of workflow engine worker threads

Parameters: wfe_num_threads

Values: Enter an integer value

Example:

wfe_num_threads=8

Workflow engine db event poll time

Purpose: To define the wait between successive queries to the DB if there is no workflow engines to run

Parameters: wfe_db_event_poll_time

Values: Enter an integer value

Example:

wfe_db_evnt_poll_time=1000

Workflow engine information

Purpose: To define the workflow engine time.

Parameters: wfl_engine_poll_time

Values: Enter an integer value

Example:

wfl_engine_poll_time=5000

Entry processor settings

Purpose: In the data entry screens, items are saved and macros run in the background. The max_threads_entry_processor should be < (db_maxConnection_appsvr_default) / 2.

Example:

max_threads_entry_processor=8

max_jobs_entry_processor=64

Event processor information

Purpose: To define the event processor time.

Parameters: evp_poll_time

Values: Enter an integer value

Example:

evp_poll_time=5000

Mount manager daemon information

Purpose: To define a unit of time before the mount manager is set to sleep.

Parameter: mountmgr_daemon_sleep_time

Values: Enter an integer value

Example:

mountmgr_daemon_sleep_time=120000

SMTP address for EmailBean

Purpose: Set the addresses for emailbean.

smtp_address=mail.company.com

from_address=support@company.com

Support email address

Purpose: Set the support email address link accessed from the Help menu. This is intended for the customer's internal support email address,

NOT the IBM WebSphere Product Center support email address.

Parameter: support_email_address

Value: Set to internal support email address

support_email_address=techsupport@company.com

Version information

Purpose: To define the WebSphere Product Center version that is installed. This value is automatically inserted during the WebSphere Product Center installation. Do not change this value.

Values: The format includes < WebSphere Product Center version number mmddyy>

Example:

version_info=4.2.0.0 041202

Character set default for the system

Purpose: Defined the default character set for the system. UTF-8 character set encoding is a requirement for WebSphere Product Center. Do not change this value.

Example:

charset_value=UTF-8

charset_name=UTF-8

Character set indicated by default in drop downs

Purpose: D efine the default character set used in drop-down fields.

Parameters: Default_charset_value, Default_charset_name

Default_charset_value=Cp1252

Default_charset_name=Default (Cp1252 - ANSI Latin 1)

Maximum item objects

Purpose: To define the maximum number of Item objects to fetch from db and to keep in memory for export in ItemSet.java

Parameters: max_fetch_value

Value: enter an integer value

Example:

max_fetch_value=100

Maximum items to merge

Purpose: To define the maximum number of items to merge and clean up.

Parameters: aggregation_queue_size

Value: enter an integer value

Example:

aggregation_queue_size=2000

Maximum Items to temporary tables

Purpose: To define the maximum number of items to occupy space in a temp tables

Parameter: aggrgation_batch_size

Value: Enter an integer value.

Example:

aggregation_batch_size=100

Maximum Items to delete/rollback

Purpose: To define the maximum number of items that can be deleted or rolled back

Parameter: delete_rollback_batch_size

Value: Enter an integer value.

Example:

delete_rollback_batch_size=100

RMI port

Purpose: To identify the RMI port used.

Parameter: rmi_port

Value: Enter the integer value of the desired rmi port used.

Example:

rmi_port=17507

Note: It is possible to bind WebSphere Product Center services to a specific rmi port by defining a variable:
<service_name>_rmi_port=<rmi_port>

I.e. To Bind 'scheduler_01' to use port '12123', define the following:

scheduler_01_rmi_port=12123

Application server variables

Purpose: To define the application server variables.

Value: Enter the integer value of the desired ports being used.

Example:

appsvr_port=7507

tomcat_ajp12=21507

tomecat_ajp13=20507

Locales setting

Purpose: To define the directory containing the per-locale XML files for language support at runtime, relative to ${TOP}.

Value: The directory defined must end with a "/". Enter a default locale if no locale is specified.

Example:

locale_xml_top=/locales/

Default locale setting

Purpose: To define the default locale if not locale is specified.

Value: Enter a locale value

Example:

default_locale=en_US

Refresh job status table

Purpose: http-equiv refresh interval for job status page. This defines how often the job status page is to be updated.

Parameter: job_status_refresh=

Value: Units are in seconds

Example:

job_status_refresh=30

Temporary directory

Purpose: Define the location of the temporary used on the local box.

Value: The directory must end with "/"

Example:

tmp_dir=/local/boxer/tmp/

Lookup dropdown box

Purpose: To define the number of values to appear in a lookup drop-down box.

Parameter: max_lookup_dropdown=

Value: Enter an integer between

Example:

max_lookup_dropdown=25

Max inactive interval

Purpose: To define the maximum time limit of inactivity before the dialog appears to disconnect from the system automatically.

Parameter: max_inactive_interval=

Value: units in seconds

Example:

max_inactive_interval=1800

Pulse update interval

Purpose: To define the time interval for the heart beat of each JVM

Parameter: pulse_update_interval

Value: Default is set to 1 minute (time in milliseconds)

Example:

pulse_update_interval=60000

Inactive JVM monitor interval

Purpose: To define the time interval for the inactivity of the JVM monitor.

Parameter: inactive_jvm_monitor_interval

Value: Default is set to 5 minutes (time in milliseconds)

Example:

inactive_jvm_monitor_interval=300000

Compressed BLOBs

Purpose: To define whether or not BLOBs are compressed.

Parameter: gzip_blobs=

Values: true/false

Example:

gzip_blobs=true

Sending outside messages

Purpose: Define whether or not outside messages can be sent

Parameter: remote_message_delivery=on

Values: on/off

Example:

remote_message_delivery=on

OS level command

Purpose: Defines the prefix used to run an OS level command

Parameter: exec_prefix

Value: Leave blank for UNIX systems.

Example:

exec_prefix=

Queue manager daemon

Purpose: The Queue Manager Daemon used by all systems is multi-threaded and by default, set to 3.

Parameter: queue_manager_threads

Value: Set default to 3.

Example:

queue_manager_threads=3

Data entry work list size

Purpose: To set the maximum size for the data entry work list that is displayed in the data entry screen. Configure this setting based on the number of concurrent users and the amount of item sets being accessed. Do not exceed the variables that appear in the example below:

Parameter: worklist_initial_size_limit,
worklist_initial_size_limit_sort_all_enabled

Value: Set and integer value

Example:

Data Entry Work List Max Size

worklist_initial_size_limit=5000

worklist_initial_size_limit_with_sort_all_enabled=500

Data Entry: Save before switching between single/multi edit

Purpose: To entries must be saved before switching between single and multi edit

Parameter: must_save_before_switching_single_multi_edit

Value: Set to true or false

> false (old behavior) union of rich search, single edit and multi edit attributes collections fetched
>
> true only applicable attributes collections fetched (lower memory footprint)

Example:

must_save_before_switching_single_multi_edit=true

Data Entry: Save entries before paging

Purpose: entry/ies must be saved before paging to the next or previous entry/ies

Parameter: must_save_before_paging_entries=true

Value: Set to true or false

> false - no saving required
>
> true -> no modified entries that are non visible (lower memory footprint)

Example:

must_save_before_paging_entries=true

Enabling inheritance

Purpose: To turn on/off inheritance functionality. Sub Specs should be enabled as well. (Inheritance not available in 4.2.1)

Parameter: enable_inheritance

Value: Set to yes or no

Example:

enable_inheritance=yes

Enabling subspecs

Purpose: To turn on/off sub specs functionality, which is used for inheritance. (Inheritance not available in 4.2.1. It is possible to use sub specs in various data modeling scenarios not related to inheritance)

Parameter: enable_subspecs

Value: Set to yes or no

Example:

enable_subspecs=yes

Queue manager service

Purpose: To define the queue manager parameters.

Parameter: queuemanager_poll_time, queuemanager_num_threads, queuemanager_max_jobs

Value: Set to an integer value

Example:

queuemanager_poll_time=5000

queuemanager_num_threads=10

queuemanager_max_jobs=1000

JMS on IBM MQ

Purpose: The following parameters are needed by implementation of JMS on IBM MQ

Parameter: jms_provider, jms_receive_timeout

Value: Set to values shown in the example below.

Example:

jms_provider=IBM WebSphere MQ

jms_receive_timeout=1000

Workflow entries editable

Purpose: Set workflow entries to be editable.

Parameter: workflow_with_entries_editable

Value: By default, workflows with entries are not editable.

no - workflows with entries are not editable

yes - workflows with entries are editable

Example:

workflow_with_entries_editable=no

JMS port and queue settings

Purpose: To set the inbound/outbound queue used by WebSphere Product Center and to set the MQ port used:

Parameters: jms_inbound_queue, jms_outbound_queue, mq_port

Example:

jms_inbound_queue=WPC.MINIME.QUEUE

jms_outbound_queue=WPC.MINIME.QUEUE

mq_port=1414

MQ settings

Purpose: The settings are needed both for implementation of MQ directly and for JMS on IBM MQ.

Parameters: mq_channel, mq_hostname, mq_queuemanager

mq_channel=WPC.JAVA.CHANNEL

mq_hostname=minime

mq_queuemanager=WPC.MINIME.QMGR

mq_inbound_queue=WPC.MINIME.QUEUE

mq_outbound_queue=WPC.MINIME.QUEUE

Specify the following parameters if the defaults are not acceptable.

mq_queue_put_open_options=

mq_message_put_options=

mq_queue_get_open_options=

mq_message_get_options=

The scripting operations can also pass these in. You have to put an integer value here. Do not comment these out, unless you have valid values that you want to override the defaults in the code. Even when these are specified, the scripting operation passed in values will override.

Character sets for messages

Purpose: Set the character set used for messages.

Parameter: mq_use_utf, mq_charset

Example:

mq_use_utf=false

mq_charset=819

Limit cache size for specs

Purpose: To set the size for static cached spec definitions displayed in WebSphere Product Center's user interface. The value affects the amount of memory used. Set the cache size according to system requirements.

Parameter: max_specs_in_cache

Value: Set integer value

Example:

max_specs_in_cache=200

Enable mount manager

Purpose: MountMgr is only useful if an external process adds/deletes files to file system directories mounted in the docstore

Parameter: enable_mountmgr

Values: true/false

enable_mountmgr=true

Set run rules per occurrence

Purpose: To enable the calculating of multi-occurrence string enumeration rule values every time. If this is not desired, set to "false".

Parameter: run_rule_per_occurrence

run_rule_per_occurrence=true

Create unique id for group/localized attributes

Purpose: Setting these parameters to true will automatically save a unique id for groups and localized attributes that are multi-occurrence. In short, this allows accurate differencing using the script operation getChangedAttributesForMultiOccurrence.

Parameters: create_unique_key_for_multi_occurrence_groups, create_unique_key_for_multi_occurrence_localized

Value: true/false

Example:

create_unique_key_for_multi_occurrence_groups=true

create_unique_key_for_multi_occurrence_localized=true

Restrict lookup tables and view objects

Purpose: Memory Settings to restrict the number of lookup tables and Container View objects that can be stored with a single session. These objects are stored with the session for performance reasons but can add to the memory footprint of each session and hence it is advisable to set a conservative number

Parameters: max_lookups_in_cache, max_ctgviews_in_cache

Values: Enter an integer value

Example:

max_lookups_in_cache=10

max_ctgviews_in_cache=10

max_roles_in_cache=50

max_accesses_in_cache=500

Enable memory monitoring

Purpose: Enable memory monitoring of session usage to analyze the memory footprint for a session.

Parameters: enable_memorymonitor, memorymonitor_interval

Values: If enabled, set to "true" enter an integer value for the interval.

Example:

enable_memorymonitor=true

memorymonitor_interval=50000

Display loading screen

Purpose: To enable/disable the display of the loading screen in between page navigation and disallows users from navigating elsewhere until the page completes loading. Set this to "true" to prevent users from flooding the server

Parameter: display_loading_screen

Values: true/false

Example:

display_loading_screen=false

Collect Profiling info for screens and jobs

Purpose: Collect profiling information for screens and jobs

Parameter: profiling_info_collection_depth=1

Values: -1 / 0 / 1 / 2 / 3 / … / 100 (-1 indicates no profiling information to be collected; depth values begin at 0)

Profiling_info_collection_depth=1

Collect additional profiling for scheduled jobs

Purpose: Collect additional profiling information for scheduled jobs

Parameter: profiling_scheduled_jobs

Values: none / nodebuglog / full

Profiling_scheduled_jobs=full

Widget names

Purpose: Use long widget names. By default, it is set to false.

Parameter: debug_use_long_widget_names

Values: true/false

debug_use_long_widget_names=false

DB connection pool interface

Purpose: ThinPoolDBManager is no longer the default DB connection pool interface layer for WebSphere Product Center. Therefore, the default value is set to "false". To switch back to using ThinPoolDBManager, set to "true"

Parameter: debug_profile_db_connections

Values: true/false

Example:

debug_profile_db_connections=true

Setting for synchronized locks

Purpose: Settings for synchronized locks, which make threads wait on the critical section until the current thread acting on the critical section

gets done with its operation. Every few moments (defined by wait_poll_time), a waiting thread polls to check if the critical section is free, and if it is free then it locks the critical sections for its use. The maximum amount of time a thread can wait on a locked critical section is specified by wait_max_time, after which an exception is thrown due to timeout.

Parameter: wait_poll_time, wait_max_time

Value: Value is entered in milliseconds. By default wait_poll_time is 1 second and wait_max_time is 1 minute.

wait_poll_time=1000

wait_max_time=60000

Set to check variables in scripts

Purpose: Set to declare variables to be declared in scripts with var. If set to true, script variables must be declared with var.

Parameter: script_check_variables

Values: true/false

script_check_variables=false

Set to trim entry attribute values

Purpose: Set to trim entry attribute values. For example "xyz " is saved as "xyz".

Parameter: trim_entry_attribute_values

Values: true/false

trim_entry_attribute_values=false

Display entry attribute list

Purpose: Set to display entry attribute lists on the Advanced Selection

Editor screen.

Parameter: display_attributes_in_rule_editor

Values: true/false

Display_attributes_in_rule_editor=true

Set immutable specs

Purpose: Set the default behavior for getCtgSpec and getCatalogSpec to get immutable or mutable specs. Immutable specs cannot be modified but reuse the shared cache. Mutable specs can be modified.

> Note: This option can be overwritten by passing a boolean to getCtgSpec/getCatalogSpec.

Parameter: get_immutable_specs

Values: true/false

Get_immutable_specs=false

Turn off event firing

Purpose: Switch event firing on and off. Setting the flag to false stops events from firing.

Parameter: fire_event_processor_events

Values: true/false

fire_event_processor_events=true

Set upper limit to nodes displayed under parent

Purpose: Set an upper limit, or not (0), to the number of node that are displayed under one parent node in the left navigation pane, also known as the catalog explorer.

Parameter: leftnav_max_categories, leftnav_max_items

Values: Enter an integer

levtnav_max_categories=0

leftnav_max_items=100

Set fully-qualified URL

Purpose: Set a fully-qualified URL, including port number, of the web site where users should point their browsers to access a WebSphere Product Center instance. It should not include a trailing "/" character. If this value is left empty, it will be deduced from the appserver hostname and port. Do not leave this value empty for WebSphere.

Parameter: trigo_web_url

Values: Enter a URL with port number

trigo_web_url=http://bach.qa.company.com:7507

File Name: admin_properties.xml

Purpose: This file is used by the administrative utilities to configure clusters of WebSphere Product Center.

Rules:

List the different hosts that are part of the cluster

Services can only be started on the machine in the cluster

Use 'localhost' if there is only one host in the cluster, otherwise use the real machine names

Example:

```
<!-- %DISCLAIMER% -->

<admin>
```

```
<!--
```

List the different hosts that are part of the cluster.

Services can only be started on the machines in the cluster.

You should only use 'localhost' if only one host is in the cluster.

Otherwise, you should use the real machine names.

Example:

```
<cluster>
<host name="server1"/>
<host name="server2"/>
<host name="server3"/>
</cluster>

<cluster>
<host name="localhost"/>
</cluster>

</admin>
```

File Name: docstore_mount.xml

Purpose: The system mount manager requires this file for the location of various OS file system mount points.

Example:

```
<?xml version="1.0"?>
<mnts>
<mnt doc_path="/public_html/" real_path="$supplier_base_dir/"/>
<mnt doc_path="/ftp/" real_path="$supplier_ftp_dir/"/>
</mnts>
```

File Name: data_entry_properties.xml

This file contains information on the additional frames that are displayed in the data entry screen when using single edit.

For a company, for a given catalog or hierarchy, it is possible to define a set of scripts that will be executed to render the additional frames.

There are two types of scripts:

- **url**: the script will consist of a function getURL(entry) which should return a url. A new iframe will be added that points to this url
- **content**: the script will consist of a function getContent(entry) which should return some html content. A new div will be added that displays this content.

For each script, you will need to specify:

type (url/content),

title

path (usually /scripts/triggers/<script name>),

some extra html information to be passed to either

iframe or the div html element (optional)

Example:

<xml>

<company code="WPC">

<catalog name="ctg1">

<script>
<type>url</type>

<extra>height='150'</extra>
<title>for ctg1 - 1</title>
<path>/scripts/triggers/test_ctg1b</path>
</script>

<script>
<type>content</type>
<title>for ctg1 - 1</title>
path>/scripts/triggers/test_ctg1b</path>
</script>

</catalog>

<hierarchy name="h1">
<script>
<type>url</type>
<title>for h2 - 1</title>
<path>/scripts/triggers/test_h2</path>
</script>
</hierarchy>

</company>
</xml>

Notices

IBM may not offer the products, services, or features discussed in this document in all countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law:

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Burlingame Laboratory
Director IBM Burlingame Laboratory
577 Airport Blvd., Suite 800
Burlingame, CA 94010
U.S.A

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement, or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurement may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not necessarily tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

This information may contain examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples may include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

Programming interface information

Programming interface information, if provided, is intended to help you create application software using this program.

General-use programming interfaces allow you to write application software that obtain the services of this program's tools.

However, this information may also contain diagnosis, modification, and tuning information. Diagnosis, modification and tuning information is provided to help you debug your application software.

Warning: Do not use this diagnosis, modification, and tuning information as a programming interface because it is subject to change.

Trademarks and service marks

The following terms are trademarks or registered trademarks of International Business Machines Corporation in the United States or other countries, or both:

IBM
the IBM logo
AIX
CrossWorlds
DB2
DB2 Universal Database
Domino
Lotus
Lotus Notes
MQIntegrator
MQSeries
Tivoli
WebSphere

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

MMX, Pentium, and ProShare are trademarks or registered trademarks of Intel Corporation in the United States, other countries, or both.

Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Other company, product or service names may be trademarks or service marks of others.

This product includes software (Axis, Jakarta Commons Collection, Jakarta Commons DBCP, Jakarta Commons Pool, Jakarta Commons Regular Expression, Log4J, Regexp, Xalan, Xerces, XML4J) developed by the Apache Software Foundation (http://www.apache.org/).

CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or

bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

a. You must give any other recipients of the Work or Derivative Works a copy of this License; and

b. You must cause any modified files to carry prominent notices stating that You changed the files; and

c. You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those

notices that do not pertain to any part of the Derivative Works; and

d. If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

CUP Parser Generator Copyright Notice, License, and Disclaimer
Copyright 1996-1999 by Scott Hudson, Frank Flannery, C. Scott Ananian

Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both the copyright notice and this permission notice and warranty disclaimer appear in supporting documentation, and that the names of the authors or their employers not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission.

The authors and their employers disclaim all warranties with regard to this software, including all implied warranties of merchantability and fitness. In no event shall the authors or their employers be liable for any special, indirect or consequential damages or any damages whatsoever resulting from loss of use, data or profits, whether in an action of contract, negligence or other tortious action, arising out of or in connection with the use or performance of this software.

JLEX COPYRIGHT NOTICE, LICENSE AND DISCLAIMER.

Copyright 1996-2003 by Elliot Joel Berk and C. Scott Ananian

Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both the copyright notice and this permission notice and warranty disclaimer appear in supporting documentation, and that the name of the authors or their employers not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission.

The authors and their employers disclaim all warranties with regard to this software, including all implied warranties of merchantability and fitness. In no event shall the authors or their employers be liable for any special, indirect or consequential damages or any damages

whatsoever resulting from loss of use, data or profits, whether in an action of contract, negligence or other tortious action, arising out of or in connection with the use or performance of this software.

Java is a trademark of Sun Microsystems, Inc. References to the Java programming language in relation to JLex are not meant to imply that Sun endorses this product.