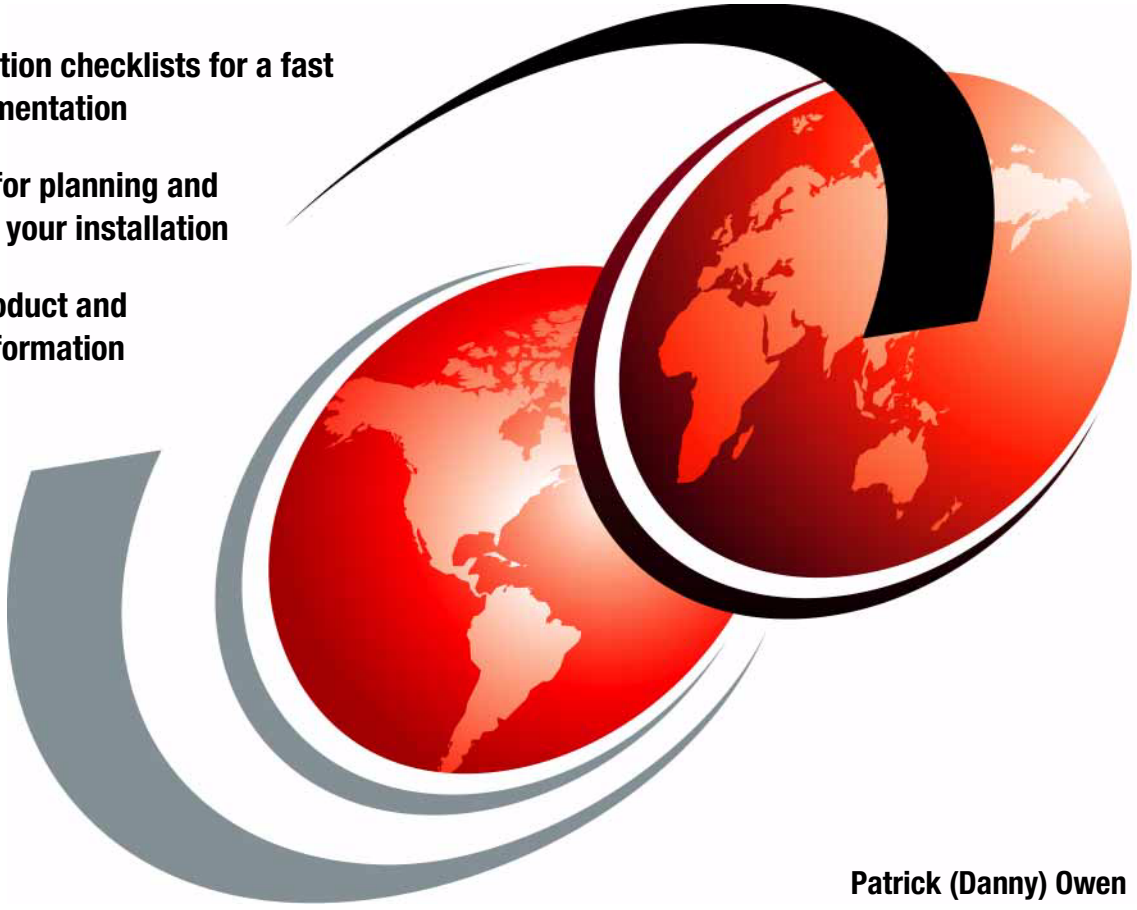
Information Management

IBM

IBM InfoSphere Information Server Installation and Configuration Guide

- Pre-installation checklists for a fast start implementation
- Guidelines for planning and configuring your installation
- Detailed product and platform information



Patrick (Danny) Owen

Redpaper

ibm.com/redbooks

International Technical Support Organization

IBM InfoSphere Information Server Installation and Configuration Guide

March 2011

REDP-4596-00

Note: Before using this information and the product it supports, read the information in “Notices” on page ix.

First Edition (March 2011)

This edition applies to Version 8.1 of IBM Information Server.

Contents

Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information about 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 the 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.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not 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 contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples 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.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs.

Trademarks

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both. These and other IBM trademarked terms are marked on their first occurrence in this information with the appropriate symbol (® or ™), indicating US registered or common law trademarks owned by IBM at the time this information was published. Such trademarks may also be registered or common law trademarks in other countries. A current list of IBM trademarks is available on the Web at http://www.ibm.com/legal/copytrade.shtml

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

AFS™
AIX®
DataStage®
DB2®
IBM®
Informix®
InfoSphere®
MVS™
Orchestrate®
Passport Advantage®
QualityStage®
RACF®
Redbooks®
Redpaper™
Redbooks (logo) ®
System z®
VisualAge®
WebSphere®
z/OS®

The following terms are trademarks of other companies:

Netezza Performance Server, Netezza, NPS, and N logo are trademarks or registered trademarks of IBM International Group B.V., an IBM Company.

Intel, Itanium, Intel logo, Intel Inside logo, and Intel Centrino logo are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

Linux is a trademark of Linus Torvalds in the United States, other countries, or both.

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

Java, and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

UNIX is a registered trademark of The Open Group in the United States and other countries.

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

Preface

This IBM® Redpaper™ publication provides suggestions, hints and tips, directions, installation steps, checklists of prerequisites, and configuration information collected from several IBM InfoSphere® Information Server experts. It is intended to minimize the time required to successfully install and configure the InfoSphere Information Server.

The information in this document is based on field experiences of experts who have implemented InfoSphere Information Server. In certain cases, the suggestions documented here might differ from the product documentation. However, except where noted, this document is intended to supplement, and not replace, the product documentation and readme files.

The primary audience for this document is administrators who have been trained on InfoSphere Information Server. The information in some sections might also be relevant for technical architects, system administrators, and developers.

The team who wrote this paper

This paper was produced by the following author, along with contributions from several of his colleagues.

Patrick (Danny) Owen has been a Field Engineer with the Center Of Excellence for Information Server since 2003. He specializes in complex installs, grid, high availability, and performance for complex and advanced needs customers. Danny has published in the field of computer science on topics such as optical character recognition and algorithms for mapping water vapor on the moon. He graduated from the University of Arkansas at Little Rock with a Bachelor of Science degree in computer science.

Thanks to the following people who contributed to the development and publication of this paper:

Chuck Ballard, Project Manager
Mary Comianos, Publications Management
Emma Jacobs, Graphics
IBM San Jose, CA

Now you can become a published author, too!

Here's an opportunity to spotlight your skills, grow your career, and become a published author—all at the same time! Join an ITSO residency project and help write a book in your area of expertise, while honing your experience using leading-edge technologies. Your efforts will help to increase product acceptance and customer satisfaction, as you expand your network of technical contacts and relationships. Residencies run from two to six weeks in length, and you can participate either in person or as a remote resident working from your home base.

Find out more about the residency program, browse the residency index, and apply online at:

ibm.com/redbooks/residencies.html

Comments welcome

Your comments are important to us!

We want our papers to be as helpful as possible. Send us your comments about this paper or other IBM Redbooks® publications in one of the following ways:

- Use the online **Contact us** review Redbooks form found at:

  ibm.com/redbooks

- Send your comments in an email to:

  redbooks@us.ibm.com

- Mail your comments to:

  IBM Corporation, International Technical Support Organization
  Dept. HYTD Mail Station P099
  2455 South Road
  Poughkeepsie, NY 12601-5400

Stay connected to IBM Redbooks

- Find us on Facebook:

  http://www.facebook.com/IBMRedbooks

- Follow us on Twitter:

  http://twitter.com/ibmredbooks

- Look for us on LinkedIn:

  http://www.linkedin.com/groups?home=&gid=2130806

- Explore new Redbooks publications, residencies, and workshops with the IBM Redbooks weekly newsletter:

  https://www.redbooks.ibm.com/Redbooks.nsf/subscribe?OpenForm

- Stay current on recent Redbooks publications with RSS Feeds:

  http://www.redbooks.ibm.com/rss.html

1

Introduction

This IBM Redbooks publication highlights various IBM InfoSphere Information Server installation topologies, platform-specific requirements, prerequisites, checklists, hardware resource suggestions, hardware configurations, I/O configuration suggestions, user and group management, InfoSphere Information Server Architecture, and post-install configurations. Our purpose is to enable an administrator, with some experience with InfoSphere Information Server, to choose a deployment topology and IS configuration to meet the needs of their enterprise, and to be successful in deploying InfoSphere Information Server.

Towards that end, we strongly suggest that you read this document in its entirety before making final choices. This will give the best foundation for ensuring that the choices are made with as broad an understanding and informed consideration as possible.

The primary audience for this document is administrators who have been trained on InfoSphere Information Server. The information in some sections might also be relevant for technical architects, system administrators, and developers.

The following sections of this chapter start by describing the platforms, environments, and specifications for the IBM InfoSphere Information Server.

1.1 IBM services offerings

IBM Information Management provides a broad set of services offerings designed to maximize success with the Information Management product suite through standard practices that have been developed across numerous successful deployments. Intended to establish a foundation of product knowledge and guidelines, these strategic workshops are tailored to the customer's existing environments, standards, and methodologies.

Within the overall project life cycle, a comprehensive set of services offerings is available using the entire Information Management suite. Complete details on each offering are available through IBM Information Management Services:

http://www.ibm.com/software/data/

1.2 Platform specification

The information in this section provides guidelines based on field experiences. In certain cases the suggestions provided here are not the same as those in the installation documentation. The installation documentation typically deals with minimum requirements, whereas the numbers contained below are based on best practice experience and will typically yield more satisfactory performance than configurations based on the minimum requirements. All platforms have additional considerations that are documented in the *DataStage Install and Upgrade Guide* and readme files.

1.2.1 Client: Windows platform specification

The IBM InfoSphere DataStage® client is tightly coupled to the DataStage server. Despite what might be indicated in the release notes, the DataStage client and server versions should always match unless you have been directed otherwise by support.

By installing the DataStage Multi-Client Manager on the client workstation, you can maintain multiple DataStage client versions on a single machine.

Suggested 32-bit operating system versions

The following 32-bit operating system versions are suggested:

- Windows XP Professional Service Pack 2
- Windows Vista Business, Windows Vista Ultimate, and Windows Vista Enterprise
- Windows Server 2003 Service Pack 2

Suggested web browsers

The following web browsers are suggested:

- Microsoft Internet Explorer 6 Service Pack 2
- Microsoft Internet Explorer 7
- Mozilla Firefox 2

Required assets

The following assets are required:

- A screen resolution of 1024x768 or better with True Color (24-bit)
- .NET framework v1.1 (included in the DataStage Client Install CD if not already installed)

Suggested memory

A minimum of 2 GB memory is suggested. Additional memory is beneficial as the size of the flows or the number of columns being processed increases.

Required disk space

The following disk space for InfoSphere Information Server Client Products is required:

- 10 MB in `\Windows\System32`
- 840 MB `\IBM\Information Server`

1.2.2 Server: Windows platform specification

DataStage for Windows release 8.1 requires a PC with an Intel processor (or equivalent) running 32-bit Windows Server 2003 Service Pack 2.

Suggested memory

The following memory is suggested:

- The minimum amount of memory for installing the InfoSphere Information Server client tier is 2 GB.
- The minimum amount of memory for installing the InfoSphere Information Server services and engine tiers on the same computer, or on separate computers, is 4 GB.

Memory requirements depend on the type of processing, the volume of parallel processing, the size and number of simultaneously running InfoSphere DataStage and IBM InfoSphere QualityStage® jobs, and the memory requirements of the operating system and other applications (such as relational databases).

Evaluate the following factors to determine the memory requirements for your system:

- The number of InfoSphere Information Server product components on the same server
- Other software on the same server
- Performance requirements
- Size and complexity of your configuration
- Extent of activity and the number of concurrent clients that access your system

Required disk space

The following disk space is required:

- 2.6 GB for InfoSphere Application Server
- 1.0 GB for IBM DB2®
- 1.4 GB for the InfoSphere Information Server components
- 2.5 GB for the metadata repository database
- 1.5 GB for the InfoSphere Information Analyzer analysis database
- 2 GB of temporary space during the installation

C++ compiler

On development systems, a C++ compiler is required to compile jobs with parallel transformers:

- Microsoft Visual C++ .NET 2003
- Microsoft Visual Studio 2005 Professional Edition C++
- Microsoft Visual Studio .NET 2005 Express Edition C++

Embedded MKS OEM

DataStage for Windows installations (and requires) a special OEM version of the MKS Framework that provides UNIX-style compatibility through runtime libraries, scripts, and utilities.

Important: The OEM version of MKS that is installed with DataStage for Windows includes make, as well as the header files and libraries necessary to build custom components, BuildOps, and transformers. As such, the DataStage MKS OEM distribution is different from OEM MKS installations included with other products, such as InfoSphere QualityStage.

1.2.3 Server: UNIX and Linux platform specification

The DataStage server component runs on the following platforms that are supported by release 8.1:

- IBM AIX® 5.3, 6.1
- HP-UX (PA-RISC) 11i v2 (11.23), 11iv3
- HP-UX (Itanium) 11i v2 (11.23), 11iv3
- Linux: Red Hat Enterprise Linux Advanced Server 4 on AMD or Intel
- Linux: Red Hat Enterprise Linux Advanced Platform 5 on AMD or Intel
- Linux: SUSE Linux Enterprise Server 10 on AMD or Intel
- Linux: SUSE Linux Enterprise Server 10 on IBM System z®
- Sun Solaris 9 and 10

Suggested memory

As with any configuration, actual memory requirements depend on the type of processing, degree of parallelism, size and number of simultaneously running DataStage jobs, and memory requirements by the operating system and other applications (such as relational databases).

The following memory suggestions are for DataStage only:

- For 4 - 16 processors: Two GB of memory per processor is generally adequate, but 3 GB is preferred for high-performance environments.
- For 16 or more processors: Less than 2 GB of memory per processor is needed except for instances with DataStage jobs that use very large lookups or hash aggregators, or when running large, complex DataStage jobs simultaneously.

Required disk space

The following disk space is required:

- 2.6 GB for InfoSphere Application Server
- 1.0 GB for DB2
- 1.4 GB for the InfoSphere Information Server components
- 2.5 GB for the metadata repository database
- 1.5 GB for the InfoSphere Information Analyzer analysis database
- 2 GB of temporary space during the installation
- 100 MB per project for design metadata and logs (might grow significantly over time)
- 25 MB of free space in `/var`
- Sufficient storage space for any data that is to be held in DataStage tables or files
- Additional space to allow for temporary data storage while a DataStage job is running

C++ compiler

On development systems, a C++ compiler is required to compile jobs with parallel transformers. When installing the C++ compiler for your machine, ensure that all packages are installed.

Important: Only the compilers and versions in Table 1-1 are compatible with DataStage. IBM certifies DataStage for specific compiler releases for a given platform.

Table 1-1 lists the supported compilers. [1]

Table 1-1 Supported compilers

Operating system	C++ compilers	Runtime components and additional requirements
64-bit AIX 5.3 64-bit AIX 6.1	May 2008 XL C/C++ Enterprise Edition V8.0 for AIX PTF, Reference #4019338 August 2008 XL C/C++ Enterprise Edition V9.0 for AIX, Reference #4020144 October 2008 XL C/C++ Enterprise Edition V10.1 for AIX, Reference #4021392	The runtime libraries are installed with the operating system.
HP-UX on PA-RISC	aCC: HP ANSI C++ B3910B A.03.85	The runtime libraries are installed with the operating system.
64-bit HP-UX 11i v2 on Intel Itanium 64-bit HP-UX 11i v3 on Intel Itanium	6.16 aCC: HP C/C++ B3910B A.06.14 6.16 aCC: HP C/C++ B3910B A.06.20	The runtime libraries are installed with the operating system.
32-bit Red Hat Enterprise Linux Advanced Server 4	acc 3.4	The runtime libraries are installed with the operating system.
64-bit Red Hat Enterprise Linux Advanced Server 4	acc 3.4.6	Available on the Red Hat Linux installation media: glibc-devel-2.3.4-2.25.i386.rpm.
32-bit and 64-bit Red Hat Enterprise Linux 5 Advanced Platform	acc 4.1.2	The runtime libraries are installed with the operating system.
64-bit Solaris 9 64-bit Solaris 10	Sun Studio 10, 11, or 12	The runtime libraries are installed with the operating system.
32-bit and 64-bit SUSE Linux Enterprise Server 10	acc 4.1.2	The runtime libraries are installed with the operating system.

[1] IBM InfoSphere Information Server at:
http://www.ibm.com/software/data/infosphere/info-server/overview/requirements.html

1.2.4 Server: z/OS platform specification

To install the parallel engine, your IBM z/OS® system should meet the following hardware and software requirements:

- Red Hat Enterprise Linux 5 Advanced Platform on IBM System z.
- SUSE Linux Enterprise Server 10 on IBM System z.
- IBM z/800, 2-way processor (2066-0X2), or an LPAR that is equivalent or bigger than that.
- z/OS Version 1.3 and later.

  To determine this, use the DISPLAY IPLINFO command and note the RELEASE value.
- IBM C/C++ compiler 1.3.
- Java 1.4 or greater.
- Review APARs OA06361 and OA07784 for applicability to your system.
- z/OS 1.3 requires UQ77835, z/OS 1.4 requires UQ77836, and DB2 v8.x requires UQ89056.

Required memory

The required memory is 2 GB or more in the LPAR. To determine this, use the **`DISPLAY M`** command and look for the line that says `HIGH REAL STORAGE ADDRESS IS` *`nnnn`*`M`. *nnnn* should be 2048 or more.

The following disk space is required:

- 250 MB of free disk space for product installation.
- 100 MB per project for design metadata and logs (might grow significantly over time)
- At least 500 MB of scratch disk space per processing node
- Sufficient storage space for any data that is to be held in DataStage tables or files

C++ compiler

A C++ compiler is required to compile jobs with parallel transformers. When installing the C++ compiler for your machine, ensure that all packages are installed. Note the directory where the compiler is installed, because you will need it for system configuration: IBM C/C++ compiler Version 1 Release 2 or later.

Note: You can install anywhere on your UNIX System Services machine, but do not install at a mount point because the installation attempts to rename the installation directory to support subsequent maintenance and upgrades. If you do attempt to install at a mount point, the installation will still work, but you will receive warning messages that might be confusing. The top-level directory is subsequently identified by the environment variable $APT_ORCHHOME.

For more information about detailed requirements, go to the IBM InfoSphere Information Server, Version 8.1, system requirements page at:

http://www-01.ibm.com/support/docview.wss?rs=14&uid=swg21315971

The installer must verify requirements, because this page is updated whenever omissions are discovered.

2

Platform specifications

The information in this section provides guidelines based on field experiences. In certain cases the suggestions documented here are not the same as those in the installation documentation. The installation documentation typically deals with minimum requirements, whereas the numbers contained below are based on best practice experience and will typically yield more satisfactory performance than configurations based on the minimum requirements. All platforms have additional considerations that are documented in the *DataStage Install and Upgrade Guide* and readme files that are available with the product documentation.

2.1 Client: Windows platform specification

In this section we describe the Windows platform specification.

Note: The DataStage client is tightly coupled to the DataStage server. Despite what might be indicated in the release notes, the DataStage client and server versions should always match unless you have been directed otherwise by support.

Installing the DataStage Multi-Client Manager on the client workstation allows you to maintain multiple DataStage client versions on a single machine.

2.1.1 Suggested 32-bit version operating systems

The suggested 32-bit version operating systems are:

- Windows XP Professional Service Pack 2
- Windows Vista Business, Windows Vista Ultimate, and Windows Vista Enterprise
- Windows Server 2003 Service Pack 2

2.1.2 Suggested web browsers

The suggested web browsers are:

- Microsoft Internet Explorer 6 Service Pack 2
- Microsoft Internet Explorer 7
- Mozilla Firefox 2

2.1.3 Required assets

The required assets are:

- A screen resolution of 1024x768 or better is suggested with True Color (24-bit).
- .NET framework v1.1 (included in the DataStage Client Install CD if not already installed).

2.1.4 Suggested memory

A minimum of 2 GB memory is suggested. Additional memory is beneficial as the size of the flows or the number of columns being processed increases.

Required disk space

The required disk space is:

- InfoSphere Information Server Client Products: 10 MB in `\Windows\System32`
- About 1 GB
- InfoSphere Information Server Business Glossary Anywhere: 3 MB

2.2 Server: Windows platform specification

DataStage for Windows release 8.1 requires a PC with an Intel processor (or equivalent) running 32-bit Windows Server 2003 Service Pack 2.

2.2.1 Suggested memory

The suggested memory is:

- The minimum amount of memory for installing the IBM InfoSphere Information Server client tier is 2 GB.
- The minimum amount of memory for installing the InfoSphere Information Server services and engine tiers on the same computer, or on separate computers, is 4 GB.

Memory requirements depend on the type of processing, the volume of parallel processing, the size and number of simultaneously running InfoSphere DataStage and InfoSphere QualityStage jobs, and the memory requirements of the operating system and other applications (such as relational databases).

Evaluate the following factors to determine the memory requirements for your system:

- The number of InfoSphere Information Server product components on the same server
- Other software on the same server
- Performance requirements
- Size and complexity of your configuration
- Extent of activity and the number of concurrent clients that access your system

2.2.2 Required disk space

The required disk space is:

- 2.6 GB for InfoSphere Application Server
- 1.0 GB for DB2
- 1.4 GB for the InfoSphere Information Server components
- 2.5 GB for the metadata repository database
- 1.5 GB for the InfoSphere Information Analyzer analysis database
- 2 GB of temporary space during the installation

2.2.3 C++ compiler

On development systems, a C++ compiler is required to compile jobs with parallel transformers:

- Microsoft Visual C++ .NET 2003
- Microsoft Visual Studio 2005 Professional Edition C++
- Microsoft Visual Studio .NET 2005 Express Edition C++

2.2.4 Embedded MKS OEM

DataStage for Windows installs (and requires) a special OEM version of the MKS Framework that provides UNIX-style compatibility through runtime libraries, scripts, and utilities.

Important: The OEM version of MKS that is installed with DataStage (for Windows) includes make, as well as the header files and libraries necessary to build custom components, BuildOps, and transformers. As such, the DataStage MKS OEM distribution is different from OEM MKS installations included with other products, such as InfoSphere QualityStage.

2.3 Server: UNIX/Linux Platform specification

The DataStage server component runs on the following platforms supported by release 8.1:

- IBM AIX 5.3, 6.1
- HP-UX (PA-RISC) 11i v2 (11.23), 11iv3
- HP-UX (Itanium) 11i v2 (11.23), 11iv3
- Linux: Red Hat Enterprise Linux Advanced Server 4 on AMD or Intel
- Linux: Red Hat Enterprise Linux Advanced Platform 5 on AMD or Intel

- Linux: SUSE Linux Enterprise Server 10 on AMD or Intel
- Linux: SUSE Linux Enterprise Server 10 on IBM System z
- Sun Solaris 9 and 10

2.3.1 Suggested memory

As with any configuration, actual memory requirements depend on the type of processing, degree of parallelism, size and number of simultaneously running DataStage jobs, and memory requirements by the operating system and other applications (such as relational databases).

The following memory suggestions are for DataStage only:

- For 4 - 16 processors: Two GB of memory per processor is generally adequate, but 3 GB is preferred for high-performance environments.
- For 16 or more processors: Less than 2 GB of memory per processor is needed except for instances with DataStage jobs that use very large lookups or hash aggregators, or when running large, complex DataStage jobs simultaneously.

2.3.2 Required disk space

The required disk space is:

- 2.6 GB for InfoSphere Application Server
- 1.0 GB for DB2
- 1.4 GB for the InfoSphere Information Server components
- 2.5 GB for the metadata repository database
- 1.5 GB for the InfoSphere Information Analyzer analysis database
- 2 GB of temporary space during the installation
- 100 MB per project for design metadata and logs (might grow significantly over time)
- 25 MB of free space in `/var`
- Sufficient storage space for any data that is to be held in DataStage tables or files
- Additional space to allow for temporary data storage while a DataStage job is running

2.3.3 C++ compiler

On development systems, a C++ compiler is required to compile jobs with parallel transformers. When installing the C++ compiler for your machine, ensure that all packages are installed.

Important: Only the following compilers and versions are compatible with DataStage. IBM certifies DataStage for specific compiler releases for a platform.

Table 2-1 shows the list of supported compilers, which was compiled from:

http://www.ibm.com/software/data/infosphere/info-server/overview/requirements.html

Table 2-1 Supported compilers

Operating system	C++ compilers	Runtime components and additional requirements
64-bit AIX 5.3 64-bit AIX 6.1	May 2008 XL C/C++ Enterprise Edition V8.0 for AIX PTF, Reference #4019338 August 2008 XL C/C++ Enterprise Edition V9.0 for AIX, Reference #4020144 October 2008 XL C/C++ Enterprise Edition V10.1 for AIX, Reference #4021392	The runtime libraries are installed with the operating system.
HP-UX on PA-RISC	aCC: HP ANSI C++ B3910B A.03.85	The runtime libraries are installed with the operating system.
64-bit HP-UX 11i v2 on Intel Itanium 64-bit HP-UX 11i v3 on Intel Itanium	6.16 aCC: HP C/C++ B3910B A.06.14 6.16 aCC: HP C/C++ B3910B A.06.20	The runtime libraries are installed with the operating system.
32-bit Red Hat Enterprise Linux Advanced Server 4	gcc 3.4	The runtime libraries are installed with the operating system.
64-bit Red Hat Enterprise Linux Advanced Server 4	gcc 3.4.6	Available on the Red Hat Linux installation media: glibc-devel-2.3.4-2.25.i386.rpm.

Operating system	C++ compilers	Runtime components and additional requirements
32-bit and 64-bit Red Hat Enterprise Linux 5 Advanced Platform	gcc 4.1.2	The runtime libraries are installed with the operating system.
64-bit Solaris 9 64-bit Solaris 10	Sun Studio 10, 11 or 12	The runtime libraries are installed with the operating system.
32-bit and 64-bit SUSE Linux Enterprise Server 10	gcc 4.1.2	The runtime libraries are installed with the operating system.

2.4 Server: z/OS platform specification

To install the parallel engine, your z/OS system should meet the following hardware and software requirements:

- Red Hat Enterprise Linux 5 Advanced Platform on IBM System z.
- SUSE Linux Enterprise Server 10 on IBM System z.
- IBM z/800, two-way processor (2066-0X2), or an LPAR that is equivalent to or larger than that.
- z/OS Version 1.3 and later.

  To determine this, use the DISPLAY IPLINFO command and note the RELEASE value.
- IBM C/C++ Compiler 1.3.
- Java 1.4 or later.
- Review APARs OA06361 and OA07784 for applicability to your system.
- z/OS 1.3 requires UQ77835, z/OS 1.4 requires UQ77836, and DB2 v8.x requires UQ89056.

2.4.1 Required memory

The required memory is 2 GB of memory or more in the LPAR. To determine this, use the DISPLAY M command and look for the line that says `HIGH REAL STORAGE ADDRESS IS` *`nnnn`*`M`. *nnnn* should be 2048 or more.

2.4.2 Required disk space

The required disk space is:

- 250 MB of free disk space for product installation
- 100 MB per project for design metadata and logs (might grow significantly over time)
- At least 500 MB of scratch disk space per processing node
- Sufficient storage space for any data that is to be held in DataStage tables or files

2.4.3 C++ compiler

A C++ compiler is required to compile jobs with parallel transformers. When installing the C++ compiler for your machine, ensure that all packages are installed. Note the directory where the compiler is installed, as it will be needed for system configuration: IBM C/C++ Compiler Version 1 Release 2 or later.

Note: You can install anywhere on your UNIX System Services machine, but do not install at a mount point because the installation attempts to rename the installation directory to support subsequent maintenance and upgrades. If you do attempt to install at a mount point, the installation still works, but you receive warning messages that might be confusing. The top-level directory is subsequently identified by the environment variable $APT_ORCHHOME.

For more information about detailed requirements, see IBM InfoSphere Information Server, Version 8.1, system requirements at:

http://www.ibm.com/support/docview.wss?rs=14&uid=swg21315971

The installer must verify the requirements, because this page is updated whenever omissions are discovered.

3

Capacity planning

With any application, the most accurate sizing is determined from real-world measurements of the actual developed job flows running on the target platform. Unfortunately, this luxury is not always available, so educated estimates must be used based on prior experience and available information, with certain elements (such as disk usage) more determinant than others. This section is intended to provide rough estimates and guidelines for sizing a DataStage environment.

3.1 Minimums

For all but the smallest of applications, a minimum production configuration consists of:

- Four processors.
- Two GB of real memory per processor, although 3 GB is suggested. With more memory-intensive applications, 4 GB is better.
- Swap space should be set to at least twice the real memory.
- Disk space (see 3.5, “Disk” on page 23).

3.2 Processor

Consider a 4-processor system as a minimum for a production InfoSphere DataStage environment. Two processors are sufficient for development and certain testing environments, including quality assurance. But if you are not prepared to put four processors to work on your production data, you will likely never realize the benefits from a parallel processing engine.

Next consider the type of processing that you will be doing. Aggregations, decimal calculations, and complex data parsing are generally considered more CPU intensive than sorts and string manipulations. So taking into account what type of work the data flows are going to be doing can be helpful in deciding whether to recommend more memory versus spending the same dollars on disk or processor.

For all but the smallest systems, add processors in groups of four. For example, 8-way or 12-way systems are suggested, but for a 6-processor system (with budgetary constraints), the money is better spent on additional memory or upgrading the I/O subsystem.

3.3 Memory

For all but the smallest deployments, consider 3 GB per processor as a minimum. One of the main benefits of parallelism is to avoid landing the data to the disk subsystem, thereby avoiding the time delay of an I/O transaction. For data flows that are going to require heavy use of sorts and partitioning (such as RDBMS sources or targets, SCD type data flows) consider 4 GB of real memory per processor.

On systems with a large number of processors (16 or more), actual memory requirements might be less than these guidelines. Consider that DataStage is a 32-bit application, so that most processes (with the exception of certain operators such as lookup and hash aggregator) are limited to 2 GB of memory per process. In these large systems, less than 2 GB of memory per processor should be adequate, unless job processing includes very large lookups or hash aggregators, or if running large complex DataStage jobs simultaneously.

As an example, let us consider the impact of sorting on a particular data flow and how memory could affect the outcome. Consider system *alpha*: four processors, 8 GB memory, 100 GB disk working with file sizes of approximately 10 GB for processing. During a relatively simple data flow we are able to alter the amount of real memory consumed by the sort stage. We are able to allocate approximately 6 GB of real system memory for sorting. So at any given moment we are swapping out 4 GB of the data file to disk to perform the sorting activity.

Now consider system *beta*: four processors, 12 GB memory, 100 GB disk working with the same files sizes. During a relatively simple data flow we are able to allocate 10 GB of real memory to the sorting operation. This means that effectively, we do not land any data to disk during the sort operations. The difference in performance of these two data systems running these data flows will be tremendous.

Remember, however, that sort operations are performed in parallel, and partitioned. Because sort memory is allocated per partition, the amount of memory required for an in-memory sort is dependent on partition skew (how data is distributed across partitions). If data is not evenly distributed across partitions, it might be necessary to allocate memory for the largest partition. Total sort memory requirements depend on the degree of parallelism multiplied by the configured sort memory usage.

Lookups are another area of consideration for memory. For non-database *normal* lookups, each reference link is loaded into shared memory, which cannot be swapped on most operating systems. Also, because normal lookups allocate memory in a single shared memory block, a contiguous free block of the required size must be available in shared memory. If the server is shared with other systems (for example, databases), then shared memory might become segmented.

As with any configuration, actual memory requirements depend on the type of processing, degree of parallelism, size and number of simultaneously running DataStage jobs, and memory requirements by the operating system and other applications (such as relational databases). In general, total memory requirements depend on the number of:

- Processes generated by the job at run time, and the:
  - Size of the job (number of stages)
  - Degree of operator combination
  - Degree of parallelism (config file, node pool assignments)
- Buffer operators and buffer size (default is 3 MB/partition/buffer)
- Simultaneously active sorts (default is 20 MB/sort/partition)
- Lookups (depends on combined size of all in-memory reference tables)
- Hash aggregators (2 K per unique key column value/calculation)
- Jobs running simultaneously (maximum requirement based on job schedule)

3.4 Swap space

In most UNIX environments, swap is allocated as needed, and is generally set to 1.5x to 2x physical memory. Certain UNIX environments calculate swap space based on the incremental not complete size. Your system administrator will have this information to assist in sizing.

Solaris allocates swap space differently from other UNIX platforms. Swap space is preallocated for every child process using the memory requirements of the parent process (and correspondingly each shared library used). Because each parallel engine job executes as a hierarchical group of processes at run time, swap space requirements on Solaris will depend on the number of simultaneous DataStage jobs and their degree of complexity. On Solaris platforms only, start with 8 GB per processor for swap allocation. Following this guideline, 128 GB of swap space would be suggested for a 16 processor system. However, when running a large number of jobs simultaneously, or when running very large jobs, the swap requirements might be greater than this starting point.

Guidelines for minimizing the number of processes generated at run time are given in the Dataflow Design Standard Practice.

3.5 Disk

One possible suggestion is a disk subsystem that has four main areas. The four main areas form a traditional batch model for data processing wherein data moved in from a flat file and out to a flat file is processed more efficiently. When reading to or writing from some other type of source/target, this design would need to be modified. Consider:

- Staging for input files
- Staging for output files
- Scratch/sort work areas
- Resource areas for the *safe* storage of parallel data sets

Note: Storage considerations for the last two items (scratch/sort and resource) is dependent on the file systems identified in the parallel configuration file used at run time. For more information about building and tuning parallel configuration files, see Chapter 5, “Parallel configuration files” on page 123.

Often, we can use a tried-and-true method of estimating the space required for each class of storage developed from experience with databases: raw-data volume times 2.5. To use this method, you must have at least a partial inventory of the data that you intend to store, its frequency of use, and its retention period. More detailed internal DataSet sizing information is available in the *DataStage Parallel Job Advanced Developer's Guide*. Keep in mind that experience with your data and implementation will cause adjustments to the formulas.

3.5.1 Staging disk for input and output files

The staging areas for input and output files should be able to contain at least two of the largest expected input/output files, plus an additional 25 - 35% for growth over time. This suggestion stems from experience in such environments where it is sometimes necessary to rerun a previous file and still have room to contain today's processing data. To guard against disk failure, redundant storage (for example, RAID 5) is suggested.

3.5.2 Scratch/sort work areas

The scratch/sort areas have several unique characteristics. First, it is optimum if they are local to the system where the engine is being run. However, it is suggested that you not secure these disk areas, as the RAID penalty is often more of a hindrance to good performance than the benefit of securing temporary files that only exist during the job run. The following formulas are rough estimates

only. There are more exact formulas, but they usually require information that is not available during the install phase of an implementation.

The size of each scratch/sort area should be as follows. For each simultaneous sort use (*X* x 1.35)/*N*, where:

- *X* is the size of the largest data file.
- *N* is the degree of parallelism used by most jobs.

For example, consider a 4 GB input file on an 8-way system. The calculation would be (4 GB x 1.35)/8 processor = 675 MB for each scratch/sort space, or (4 x 1.35)/8 per partition, and there are eight in this case.

When there is more than one simultaneous sort (within multiple branches of one job or across separate, simultaneously running jobs), the total sort requirements will be the sum of each sort.

3.5.3 Resource areas for parallel data sets

Parallel data sets are most often used for intermediate storage of data between jobs (end-to-end parallelism) and for checkpoint/restart. This area's size can be calculated in roughly the same manner as the scratch/sort areas with the exception that it might be necessary to store more than one version of any given data set. For example, it might be necessary to have the previous night's data set for delta processing, or some number of prior runs for recovery purposes. Remember that since data sets are persistent, there needs to be allocated storage space for all data sets, not just those needed for the currently running job. The formulas that follow are rough estimates only. Exact formulas are available, but usually require information that is not available during the install phase of an implementation.

Therefore, the calculation for each resource disk area should be (*X* x 1.35)/*N*, where:

- *X* is the total size of all data to be stored concurrently.
- *N* is the number of processors expected for most jobs that run in parallel.

Using our example job only and storing a week's worth of data for a single job only, there will be (4 GB x 5 copies x 1.35) / 8 processor = ~3.38 GB per partition. So, in this case, there will be eight partitions.

For a more detailed estimation for the size of a parallel data set, see “Estimating the size of a parallel data set” on page 146.

4

Installation and configuration

This chapter provides information about installing and configuring IBM InfoSphere Information Server. It includes a pre-installation checklist that describes the known factors for the five layers of an InfoSphere Information Server 8.x installation.

4.1 Pre-installation overview

IBM InfoSphere Information Server provides a unified foundation for enterprise information architectures, combining the capabilities of DataStage, QualityStage, Information Analyzer, Business Glossary, InfoSphere Information Services Director, and Metadata Workbench.

InfoSphere Information Server is installed in the following five layers:

- Client
- Metadata repository
- Domain (platform services)
- Engine
- Documentation

Product components are installed in each tier, depending on the install selections that are made.

To ensure a successful installation that meets functional and performance requirements, it is critical that overall planning and prerequisites be met for all tiers. This document provides a detailed methodology for planning an InfoSphere Information Server installation.

Important: Establishing an InfoSphere Information Server environment that meets performance expectations requires a capacity planning exercise: reviewing deployment architecture, server, disk, and network configuration, data sources, targets, data volumes, processing requirements, and service level agreements.

Although minimum system requirements are referenced in this checklist, capacity planning is outside the scope of this document.

Where possible, links are provided for additional details and reference documentation is mentioned. The information in this document is based on field experiences. In certain cases, the suggestions documented here might differ from the product documentation. Except where noted, this document is intended to supplement, not replace, the product documentation and readme files.

Complete product documentation for IBM InfoSphere Information Server is only available by installing the documentation tier of the product install. Additional and updated documentation is available through the IBM InfoSphere Information Server Information Center at:

http://publib.boulder.ibm.com/infocenter/iisinfsv/v8r0/index.jsp

4.2 Pre-installation checklist

The checklist in Table 4-1 outlines the areas that must be reviewed and the steps that must be completed prior to installing InfoSphere Information Server. A copy of this checklist is also included in “Pre-installation checklist” on page 155.

Table 4-1 Pre-installation checklist

Complete	Task
	Review release notes (InfoSphere Information Server, IBM WebSphere® Application Server, DB2).
	Review the *IBM Information Server Planning, Installation and Configuration Guide*, GC19-1048-07.
	If migrating from previous versions of DataStage or QualityStage, review *Migrating to IBM Information Server Version 8*.
	Choose and validate installation architecture/topology.
	Validate system requirements for all tiers (engine, domain, repository, client, documentation).
	Verify domain (WebSphere Application Server) requirements.
	Verify database requirements for the metadata repository.
	If applicable: Verify database requirements for Information Analyzer analysis database.
	Verify and configure disks, volume groups, and file systems.
	Verify and configure operating system and resource limits.
	Verify connectivity and network configuration.
	Configure operating system users, groups, and associated permissions.
	Verify and install C++ compiler or runtime libraries, or both.
	Verify Information Server connector requirements.
	Download and install fix pack packages (InfoSvr, WebSphere, DB2).
	Perform complete system backup.

Specific details on each step can be found in subsequent sections of this document, the release notes, and the *IBM Information Server Planning, Installation, and Configuration Guide*, GC19-1048-07.

4.3 Reviewing release notes

Release notes contain the latest information about a particular release of IBM InfoSphere Information Server including issues resolved, known issues, and workarounds. There are separate release notes that should be consulted for IBM WebSphere Application Server and (if applicable) DB2.

4.3.1 IBM InfoSphere Information Server release notes

Although a copy of the InfoSphere Information Server release notes is included with a product installation, a later version might be available online in the IBM Information Center and should be consulted before any installation. For more information, see “IBM Information Server release notes” in the IBM InfoSphere Information Server Information Center at:

http://publib.boulder.ibm.com/infocenter/iisinfsv/v8r1/topic/com.ibm.swg.im.iis.productization.iisinfsv.nav.doc/containers/cont_iisinfsrv_rnote.html

When reviewing release notes, be sure to choose the version that matches the base installation version of InfoSphere Information Server, for example, 8.1.

All InfoSphere Information Server fix packs and patches that are applied after a base installation have separate, corresponding release notes that should also be reviewed. For example, an 8.1 fix pack is applied to an existing 8.1 base installation.

4.3.2 WebSphere Application Server release notes

The domain tier of IBM InfoSphere Information Server 8.1 requires WebSphere Application Server standalone release 6.0.2.27. No other versions are supported with V8.1. On most platform configurations, WebSphere Application Server is included with the InfoSphere Information Server installer, but there are some exceptions (namely, 64-bit platforms).

Release notes for IBM WebSphere Application Server v6.0.2 are available at:

http://publib.boulder.ibm.com/infocenter/wasinfo/v6r0/topic/com.ibm.websphere.base.doc/info/aes/ae/v6rn.html

Complete documentation for WebSphere Application Server v6.0.2 is available through the Info Center at the following location:

http://publib.boulder.ibm.com/infocenter/wasinfo/v6r0/topic/com.ibm.websphere.base.doc/info/welcome_base.html

DB2 Enterprise Server Edition release notes

The InfoSphere Information Server metadata repository can be installed in DB2 Enterprise Server Edition v9, Oracle 10g R2, or Microsoft SQL Server 2005.

When installing InfoSphere Information Server, if you choose to install the *metadata server* option, a new copy of DB2 v9.1 (32 bit) or DB2 v9.5 (64 bit) is installed. You can find the release notes for DB2 v9.1 at:

http://publib.boulder.ibm.com/infocenter/db2luw/v9/topic/com.ibm.db2.udb.doc/doc/c0023859.htm

4.4 Planning, installation, and configuration

This pre-installation checklist provides a methodology and supplemental information about preparing for an installation of IBM InfoSphere Information Server:

- Planning to install IBM InfoSphere Information Server
- Preparing your system for installation
- Installing IBM InfoSphere Information Server
- Configuring IBM InfoSphere Information Server
- Installing additional components
- Troubleshooting installations
- Removing IBM InfoSphere Information Server

For details about these steps, see *IBM Information Server Planning, Installation, and Configuration Guid*e, GC19-1048-07. This guide is part of the IBM InfoSphere Information Server documentation, which is included with the product installation package. The documentation tier can be installed separately, without installing the rest of the InfoSphere Information Server.

4.5 Reviewing migrating to InfoSphere Information Server

If migrating from earlier versions of DataStage or QualityStage, review the content in the *Migrating to IBM Information Server Version 8 Guide*, which is included with the product installation package. The documentation tier can be installed separately, without installing the rest of the InfoSphere Information Server.

4.6 Choosing and validating the architecture or topology

This section explains how to choose and validate the installation architecture topology. The architecture of IBM InfoSphere Information Server is organized into four major layers, which are listed in Table 4-2.

Table 4-2 Architecture layers

Layer	Description	Architectural notes
Client	Administration, analysis, and development user interfaces and optional MetaBrokers and bridges.	Multiple clients can access a single InfoSphere Information Server. Client and server version must match. Using v8 MultiClientManager, multiple client versions can be installed on a single workstation. Requires 32-bit versions of Windows XP, Windows Vista, or Windows 2003.
Metadata repository	Database that stores InfoSphere Information Server settings, configuration, design, and runtime metadata.	A single metadata repository database is defined for each InfoSphere Information Server installation. DB2 (included or customer supplied) v9.1 for 32 bit, v9.5 for 64 bit. Oracle 10g R2 (customer supplied). SQL Server 2005 (customer supplied).
Domain	InfoSphere Information Server common and product-specific services.	A single domain is defined for each InfoSphere Information Server installation. Requires WebSphere Application Server (included or customer supplied) Release 6.0.2 Fix Pack 27 or later fix packs only. Standalone (non-network) profile only.
Engine	Runtime engine that executes all InfoSphere Information Server tasks. Includes engine, connector, PACK, and service agents (logging, ASB, JobMon, PerfMon).	Multiple engines on separate server environments might be registered in a single InfoSphere Information Server domain (cluster or grid deployment). Only one v8 engine can be installed on a single server. Can co-exist with multiple v7 DataStage engines.

In any InfoSphere Information Server installation, the release level (version + fix pack + patches) must match on all layers.

Important: Although the architecture of InfoSphere Information Server offers many theoretical deployment architectures, only the following subset of configuration tiers is suggested and supported.

Any deviation from these deployments must be reviewed by IBM Support and Engineering.

4.6.1 Two-tier deployment

Figure 4-1 illustrates a classic two-tier deployment:

- Clients: Client applications on Windows system
- Server 1: All other components (metadata repository, domain, engine) on the same Linux, UNIX, or Windows server

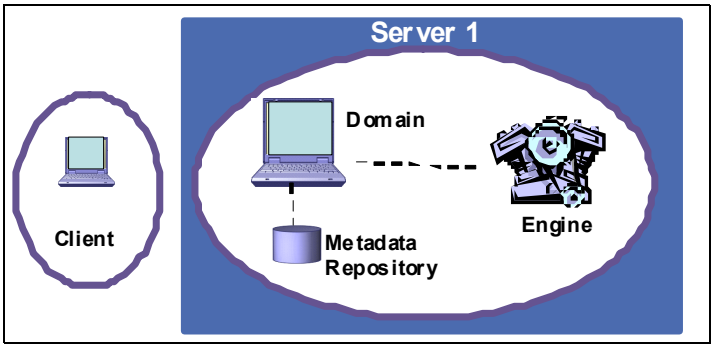


Figure 4-1 Two-tier deployment

In this configuration, the single server housing the metadata repository, domain, and engine should have a minimum of 8 GB of RAM, in addition to meeting the combined disk requirements outlined in the InfoSphere Information Server system requirements.

The InfoSphere Information Server versions of all components must match.

Because the complexity of tracking the state of different layers is simplified, the two-tier deployment is optimal for high-availability (failover) scenarios.

4.6.2 Three-tier deployment

Figure 4-2 illustrates a classic three-tier deployment:

- Clients: Client applications on Windows system
- Server 1: Metadata server (metadata repository and domain) on the same Linux, UNIX, or Windows server
- Server 2: Engine on Linux, UNIX, or Windows server

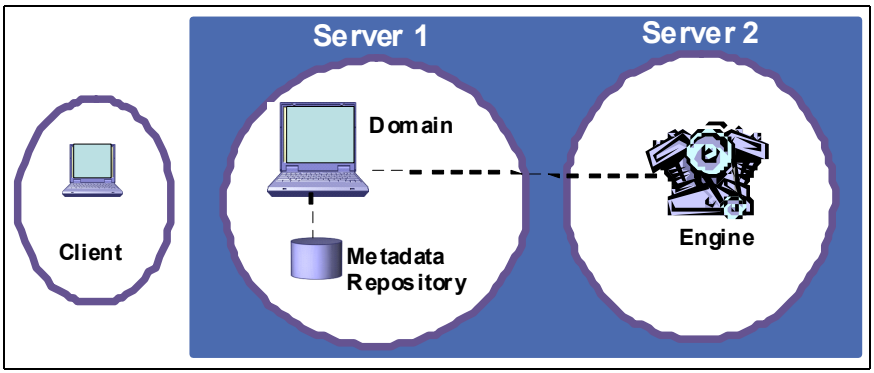


Figure 4-2 Three-tier deployment

In the three-tier configuration, both backend servers (metadata server and engine) must be located in the same physical data center, and should be connected to the same network subnet.

The operating system of the metadata server and the engine servers should be the same.

4.6.3 Four-tier deployment

The four-tier deployment, which next segments the metadata repository from the domain, has the following configuration:

- Clients: Client applications on Windows system
- Server 1: Metadata repository on Linux, UNIX, or Windows server
- Server 2: Domain on Linux, UNIX, or Windows server
- Server 3: Engine on Linux, UNIX, or Windows server

For performance reasons, this configuration is not suggested unless the metadata repository and domain servers are connected by a dedicated, private, high-speed network connection.

In the four-tier configuration, all backend servers must be located in the same physical data center, and should be connected to the same network subnet. The connection between these two machines must be low latency to avoid negatively impacting the UI environment.

4.6.4 Cluster and grid deployments

In all configurations, multiple InfoSphere Information Server (v8) engines can be installed or cross-mounted on physically separate servers. By including these servers in a parallel configuration file at run time, InfoSphere Information Server processing can span across a single server boundary.

Figure 4-3 illustrates a cluster and grid deployment.

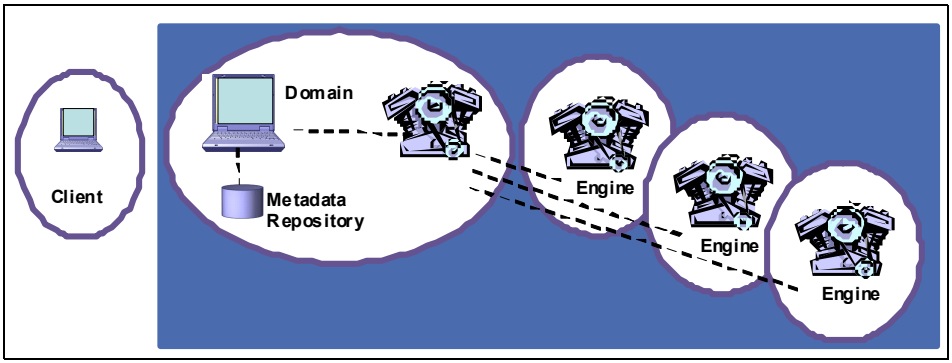


Figure 4-3 Cluster and grid deployment

In a cluster or grid deployment, all engine servers must be the same operating system, must be located in the same physical data center, and should be connected by a dedicated, private high-speed network connection.

4.6.5 Wide area network deployments

Due to the data exchanges that occur between the InfoSphere Information Server layers, it is strongly suggested that all tiers be located in the same local area network (LAN). When deploying in a wide area network (WAN) configuration, a network-hosting tool (for example, Citrix or Windows Remote Desktop) must be used to host the InfoSphere Information Server clients.

4.7 Validating system requirements for all tiers

After selecting the installation topology, IBM InfoSphere Information Server system requirements should be verified for the client, metadata repository, domain, and engine.

For more information about the IBM InfoSphere Information Server system requirements, see the IBM InfoSphere Information Server, Version 8.1, system requirements at:

http://www.ibm.com/support/docview.wss?rs=14&uid=swg21315971#dqx1windowsdiskspace

4.8 Verifying domain requirements

The domain layer requires WebSphere Application Server Release 6.0.2 Fix Pack 27 or later fix packs only (WebSphere Application Server should return 6.0.2.27 as its version). Specific hardware and software requirements for WebSphere Application Server Release 6 can be found online:

- For WebSphere Application Server 6.0.2 hardware requirements, see:

  http://www-1.ibm.com/support/docview.wss?rs=180&uid=swg27007250

- For WebSphere Application Server 6.0.2 software requirements, see:

  http://www-1.ibm.com/support/docview.wss?rs=180&uid=swg27007256

On most platforms, when you install InfoSphere Information Server, you can install a new copy of WebSphere Application Server or use an existing installation.

On certain 64-bit platforms (64-bit RedHat Linux, 64-bit SUSE Linux, HP-UX Itanium) it is necessary to download the WebSphere Application Server 6.0.2 installer and fix packs from IBM Passport Advantage®. For specific instructions that are included in the IBM InfoSphere Information Server system requirements see:

http://www-01.ibm.com/support/docview.wss?rs=14&uid=swg21315971#dqx1windowsdiskspace

To install the domain layer in an existing installation of WebSphere Application Server it must meet the following requirements:

- Be at Release 6.0.2 Fix Pack 27 or later fix pack (WebSphere Application Server 6.0.2.27).
- Be a standalone (not network) deployment.

- Use a new, empty profile.
- The profile must be named server1.

Important: To minimize risk and to ensure that the domain requirements are satisfied, use the version of WebSphere bundled with the IBM InfoSphere Information Server installation, except in specific configurations (64-bit installations of RedHat Linux, SUSE Linux, or HP-UX Itanium).

4.9 Verifying database requirements for metadata repository

The metadata repository layer can be installed on DB2 v9.1, DB2 v9.5, Oracle 10g R2, or Microsoft SQL Server 2005. By default, the metadata repository database is named *xmeta*.

When installing InfoSphere Information Server, you can install a new copy of DB2 Version 9.1 (or 9.5, depending on the target platform) or use an existing installation. If you want to use Microsoft SQL Server or Oracle, you must install and configure them before you install InfoSphere Information Server.

Specific hardware and software requirements for DB2 can be found online. For DB2 v9 requirements see the following website:

http://publib.boulder.ibm.com/infocenter/db2luw/v9/topic/com.ibm.db2.udb.uprun.doc/doc/r0025127.htm

To install the metadata repository on an existing installation of DB2, it must meet the following requirements:

- DB2 Enterprise Server Edition Release 9.1 (or 9.5)
- Database cannot be partitioned

On certain 64-bit platforms (64-bit RedHat Linux, 64-bit SUSE Linux, HP-UX Itanium) it is necessary to download the DB2 9.5 installer and fix packs from IBM Passport Advantage. Specific instructions are included in the IBM InfoSphere Information Server system requirements at:

http://www.ibm.com/support/docview.wss?rs=14&uid=swg21315971#dqx1windowsdiskspace

During the installation of InfoSphere Information Server, if you choose to install the metadata repository on a new copy of DB2, the metadata repository database will be created as part of the installation.

If you choose to install the metadata repository on an existing installation of DB2, Oracle 10g R2, or SQL Server 2005:

- The metadata repository (xmeta) database must be created before the InfoSphere Information Server installation.
- Database connectivity must be configured from the domain tier to the metadata repository database.

Database creation scripts are included in the Database Support subdirectory of the InfoSphere Information Server installation. For information about creating the metadata repository database, see *IBM Information Server Planning, Installation, and Configuration Guide*, GC19-1048-07.

Important: The InfoSphere Information Server requirements recommend a minimum of 3 GB for the metadata repository. However, you must closely monitor the database as it grows to ensure that sufficient space is available in the tablespaces and underlying file systems.

As table definitions, job designs, reports, and runtime metadata are created, the size of the metadata repository can grow significantly.

4.10 Verifying database requirements for Information Analyzer analysis

This section explains how to verify database requirements for the Information Analyzer analysis database. If deploying Information Analyzer, the Information Analyzer analysis database can be installed on DB2 v9.1 and v9.5, Oracle 10g R2, or Microsoft SQL Server 2005. By default, the Information Analyzer Analysis database is named *IADB*.

When installing InfoSphere Information Server, you can install a new copy of DB2 Version 9.1 (or 9.5) or use an existing installation. If you want to use Microsoft SQL Server or Oracle, you must install and configure them before you install InfoSphere Information Server.

If you choose to install the Information Analyzer Analysis database on an existing installation of DB2, Oracle 10g R2, or SQL Server 2005:

- The Information Analyzer Analysis database (IADB) must be created before the InfoSphere Information Server installation.
- Database connectivity must be configured from the domain tier to the Information Analyzer Analysis database.

Database creation scripts are included in the Database Support subdirectory of the InfoSphere Information Server installation. For information about creating the Information Analyzer Analysis database, see the *IBM Information Server Planning, Installation, and Configuration Guide*, GC19-1048-07.

Important: Although the InfoSphere Information Server System requirements recommend a minimum of 3 GB for the Information Analyzer Analysis database, the actual size depends on the size of the sources to be analyzed. Unless using sampled analysis, the Information Analyzer Analysis database can be larger than the combined size of all analyzed data sources.

4.11 Verifying and configure disks, volume groups, file systems

In this section we describe how to verify and configure disks, volume groups, and file systems.

4.11.1 RAID or SAN configuration

IBM InfoSphere Information Server uses file system mount points for its installation, libraries, temporary storage, and data set storage.

Ultimate performance of an InfoSphere Information Server job (DataStage, QualityStage, and Information Analyzer) depends on all components being optimized. When discussing disk (including RAID and SAN) configuration, maximum performance is a combination of maximum bandwidth (controllers, disk) and minimized contention.

The following guidelines can assist in the configuration of RAID or SAN technologies:

- Minimize contention between temporary (scratch, buffer, sort) and data file systems.
- Minimize contention between the disks and controllers associated with InfoSphere Information Server file systems and other applications or servers.
- Consider isolating multiple mount points to separate high-speed disk interconnects and controllers.
- Consider the trade-off between the granularity of file systems and underlying configuration versus available, unused storage.

- Do not create overly complex device configurations. These can be difficult to administer and use, and might not offer a corresponding performance improvement.
- If possible, test your configuration with expertise from hardware, storage, operating system, and application (InfoSphere Information Server) resources.
- Remember that previous experience in designing I/O systems for non-parallel engines might lead to non-optimal configurations for InfoSphere Information Server.

The optimal disk configuration will strike a balance between cost, complexity, ease of administration, and ultimate performance.

4.11.2 InfoSphere Information Server file systems

IBM InfoSphere Information Server requires file systems for:

- Software install directories
  - IBM InfoSphere Information Server
  - Domain (WebSphere Application Server)
  - Database server
- Database storage
  - Metadata repository (xmeta)
  - Information Analyzer analysis database
- InfoSphere Information Server project (runtime shadow repository) directories

  Used to capture runtime metadata (such as error messages and logging information) for the InfoSphere Information Server parallel engine. Synchronized to the metadata repository.
- Data file storage
  - InfoSphere Information Server engine temporary storage (scratch, temp, buffer)
  - InfoSphere Information Server artifacts (such as surrogate key files)
  - InfoSphere Information Server parallel data set segment files
  - Staging and archival storage for any source/target files

File system requirements:

- File systems should be expandable without requiring destruction and recreation.
- Local file systems should be reserved for temporary and scratch storage.

- Data and install directories should be created on high-performance shared file systems.
- Always use NFS *hard mounts*. (*Soft mounts* can lead to corruption of InfoSphere Information Server requiring a reinstall or restore from backup.)

Some components and plug-ins might require additional file system permissions. For example, the DB2 Enterprise Stage requires that the db2instance ID has read access to a subdirectory under $TMPDIR for db2load.

Important: Each storage class should be isolated in separate file systems to accommodate their various performance and capacity characteristics and backup requirements.

The default installation directories are best suited for small prototype environments.

4.11.3 Software installation directories

Table 4-3 describes the software installation directories.

Table 4-3 Software installationdirectories

Installation directory	Contents	Default UNIX path
Information Server	InfoSphere Information Server engine, libraries, communication agents, Job Monitor, Performance Monitor, Java JRE, and uninstall files	`/opt/IBM/Information Server/`
Domain	WebSphere Application Server executables, libraries: - InfoSphere Information Server shared services - InfoSphere Information Server product-specific services	`/opt/IBM/WebSphere/ AppServer/`
Database server	DB2 v9.1 database server executables, libraries	`/opt/IBM/db2/V9/`

Installing file system requirements

Activities for installing file system requirements are:

- Do not install InfoSphere Information Server components (InfoSphere Information Server, WebSphere Application Server, and DB2) on a top-level mount point. Always install InfoSphere Information Server components to a subdirectory within a mount point. The installer components change ownership and permissions of directories that are installed, and not all mount points allow these changes, causing the install to fail.

- On Windows platforms, InfoSphere Information Server cannot be installed on virtual drives (mapped or SUBST drives).
- The installation directories for WebSphere Application Server and (if applicable) DB2 must be empty. If the target directory is not empty, the installer will attempt to create another directory, which will not be properly referenced by the InfoSphere Information Server paths.
- All file systems used by InfoSphere Information Server for installation must have read and write permissions for the primary DataStage users and groups. Ensure that each level of the target install directories is set to 755 permissions.
- InfoSphere Information Server, WebSphere Application Server, and (if applicable) DB2 must be installed into separate directories. They cannot be installed in the same directory level.
- On UNIX platforms, the DB2 installer must have write access to several file systems, including `/var` and `/usr/local/bin`. This requirement is not always satisfied when these directories are mounted from network file systems such as IBM AFS™, or if these directories are configured as read-only file systems.
- For cluster or grid implementations, share the installation file systems across all servers (with the same fully qualified paths).

4.11.4 Database storage

The two types of data storage are:

- Metadata repository (xmeta)
- Information Analyzer analysis database

On UNIX platforms, using the supplied DB2 database engine, databases are created in the home directory (`/home/db2inst1/`) of the DB2 instance owner, by default.

When using an existing installation of DB2 9.1, 9.5, Oracle 10g R2, or Microsoft SQL Server 2005, table spaces are created by the database administrator using existing file systems or raw disk devices.

Important: Closely monitor the metadata repository and (if applicable) Information Analyzer Analysis databases to ensure that sufficient space is available in the tablespaces and underlying file systems as the databases grow. Perform regular, scheduled backups on these databases.

4.11.5 InfoSphere Information Server Project directories

The InfoSphere Information Server Project (runtime shadow repository) Directory is used to capture runtime metadata (such as error messages and logging information) for the InfoSphere Information Server parallel engine. This data is synchronized into the metadata repository on periodic intervals for later reporting. The InfoSphere Information Server Project Directory can also store legacy Server Edition hash files. Project directories can grow to contain thousands of files and subdirectories depending on the number of projects, the number of jobs, and the volume of logging information retained about each job.

During the installation process, the projects subdirectory is created in the InfoSphere Information Server installation directory. By default, the DataStage administrator client creates projects in the `/opt/IBM/InformationServer/Server/Projects/` subdirectory.

During installation, Information Analyzer also creates its own project within this directory.

The following are InfoSphere Information Server Project Directory guidelines:

- Do not create DataStage projects in the default directory within the installation file system, because disk space is typically limited. Create projects in their own file system.
- On most operating systems, it is possible to create file systems at non-root levels. Create a separate file system for the projects directory within the InfoSphere Information Server installation. Back up any existing projects (retaining ownership and permissions) before mounting a separate file system over the existing `/opt/IBM/InformationServer/Projects` directory.
- For cluster or grid implementations, it is generally best to share the projects file system across servers (with the same fully qualified path).

Project naming considerations

The name of an InfoSphere Information Server Project is limited to a maximum of 54 characters. The project name can contain alphanumeric characters and it can contain underscores.

Project names cannot use the following reserved words:

- ds
- DS
- uv
- UV

Project names should be maintained in unison with source code control. As projects are promoted through source control, the name of the phase and the project name should reflect the version, in the form:

<Phase>_<ProjectName>_<version>

Where *Phase* corresponds to the phase in the application development life cycle (Table 4-4).

Table 4-4 Application development lifecycle

Phase name	Phase description
dev	Development
it	Integration test
uat	User acceptance test
prod	Production

Project directory monitoring and maintenance

Effective management of space is important to the health and performance of a project. As jobs are added to a project, new directories are created in this file tree, and as jobs are run, their log entries multiply. These activities cause file system stress (for example, more time to insert or delete DataStage components, longer update times for logs). Failure to perform routine project maintenance (for example, remove obsolete jobs and manage log entries) can cause project obesity and performance issues.

Important: The project file system should be monitored to ensure that adequate free space remains. If the Project file system runs out of free space during runtime activity, the repository may become corrupted, requiring a restore from backup.

4.11.6 Dataset and Scratch directories

During installation, two directories are created within the InfoSphere Information Server installation directory tree for storage of temporary and intermediate data files used by the parallel engine:

- `/opt/IBM/InformationServer/Server/Datasets` (persistent storage between jobs)
- `/opt/IBM/InformationServer/Server/Scratch` (temp storage for sort and buffer overflow)

The InfoSphere Information Server installer creates a default parallel configuration file (`/opt/IBM/InformationServer/Configurations/default.apt`) that references the default datasets and scratch directories.

Parallel configuration files are used to assign resources (such as processing nodes, disk, and scratch file systems) at run time when a job is run by the InfoSphere Information Server engine. For more information about parallel configuration files, see the *IBM Information Server Parallel Job Developer Guide*, LC18-9891-02.

Dataset and scratch directory guidelines

The following are dataset and scratch directory guidelines:

- Dataset and scratch file systems should be created outside of the InfoSphere Information Server installation directory.
- Scratch file systems should be created on local (internal) storage.
- Dataset and install directories should be created on high-performance, shared file systems with the same fully qualified path on all engine servers. Using a shared file system also facilitates high-availability failover scenarios.
- The `default.apt` configuration file should be edited to reference the newly created dataset and scratch file systems, and to ensure that these directories are used by any other parallel configuration files.
- For optimal performance, file systems should be created in high-performance, low-contention storage.
- For best performance, and to minimize storage impact on development activities, separate file systems should be created for each data and scratch resource partition.
- On systems where multiple phases are shared on the same server, consider separating data and scratch storage to different file systems for each deployment phase to completely isolate each environment. This might be required for security compliance in some situations.

4.11.7 Extending the DataStage project for external entities

For DataStage and QualityStage environments, it is suggested that another directory structure be created to integrate all aspects of a DataStage application that are managed outside of the DataStage Projects repository. This hierarchy should include directories for secured parameter files, data set header files, custom components, IBM Orchestrate® schemas, SQL, and shell scripts. It might also be useful to store custom job logs and reports.

4.11.8 File staging

Use a separate staging file system and directory structure to store, manage, and archive various source data files.

4.12 Verifying and configuring OS and resource limits

On most platforms, InfoSphere Information Server, WebSphere Application Server, and the database server have specific operating system (kernel) and user (shell) resource requirements. For the latest versions of these requirements, go to the following web pages:

- IBM InfoSphere Information Server system requirements:

  http://www.ibm.com/support/docview.wss?rs=14&uid=swg21315971#dqx1windowsdiskspace

- WebSphere Application Server 6.0.2.27 software requirements:

  http://www.ibm.com/support/docview.wss?rs=180&uid=swg27007256

- DB2 v9.1, 9.5 requirements:

  http://publib.boulder.ibm.com/infocenter/db2luw/v9/topic/com.ibm.db2.udb.uprun.doc/doc/r0025127.htm

If using an existing Oracle10gR2 or SQL Server 2005 database server for the metadata repository or Information Analyzer analysis database, reference the database vendor's system requirements.

Notes on operating system requirements

Keep in mind the following notes about operating system requirements:

- On UNIX installations of IBM InfoSphere Information Server, the tables in the following sections list the minimum requirements for UNIX kernel parameters. These settings give the generic names for the kernel parameters. The actual name and the case vary for each UNIX platform.
- Always make a backup of the kernel settings before making these changes. On many platforms, this can be accomplished by backing up the `/etc/system` file.
- On certain platforms, it might be necessary to rebuild the kernel with changes to these parameters. All changes should only be made by a trained UNIX system administrator.
- For readability, the table values in this section include commas. Omit them when setting the actual parameters.

4.12.1 UNIX kernel parameters for all platforms

Table 4-5 lists the minimum requirements for kernel parameters on all UNIX platforms. These settings give the generic names for the kernel parameters. The actual name and the capitalization vary for each platform.

The minimum values depend on the architecture as well. For example, DB2 might require a minimum value of 1024, but the engine might require 2048. If they are both installed on the same server, you must take the higher of the two.

Note: For readability, the table values in this section include commas. Omit them when setting parameters.

Table 4-5 lists the UNIX Kernel parameters.

Table 4-5 UNIX Kernel parameters

Kernel parameter	Description	InfoSphere Information Server minimum	DB2 minimum	Notes
MAXUPROC	Maximum number of processes	100 per processor		Set to at least 100 processes per node.
NOFILES	Number of open files per process	1,000		

4.12.2 UNIX user (shell) parameters for all platforms

On all UNIX platforms, adjust the per-user (shell) parameters to ensure that the InfoSphere Information Server parallel engine has sufficient resources. Table 4-6 lists these parameters.

Table 4-6 UNIX user parameters

User parameter	Description	Suggested setting
umask	Default file permissions	022
ulimit	Maximum number of user processes	8192

4.12.3 AIX system configuration

On AIX systems, if you intend to use the DataStage Job Scheduler, you must the permissions on the `/usr/spool/cron/atjobs` directory from 770 to 775 (rwxrwxr-x).

AIX kernel parameters

Table 4-7 lists the minimum requirements for UNIX kernel parameters for AIX installations of IBM InfoSphere Information Server.

Attention: For readability, the table values in this section include commas. Omit them when setting parameters.

Table 4-7 AIX kernel parameters

Kernel parameter	Description	InfoSphere Information Server minimum	Notes
SHMMAX	Maximum shared memory segment size	536870912	If disk caching is turned on and DISKCACHE is larger than 512, this must be set higher.
SHMMNI	Shared memory identifiers	2000	
SHMSEG	Maximum number of shared memory segments per process	200	

AIX user (shell) parameters for parallel jobs

Table 4-8 lists the additional per-user (shell) parameters that should be adjusted on AIX to ensure that the InfoSphere Information Server parallel engine has sufficient resources.

Table 4-8 AIX user parameters for parallel jobs

User parameter	Description	Suggested setting
fsize	Largest file that a user can create	2 GB - 512 bytes (4,194,303 512-byte blocks)
data	Largest data segment (heap) that a program can have	128 MB minimum (262,144 512-byte blocks) or -1 for no limit

User parameter	Description	Suggested setting
stack	Largest stack size that a program can have	32 MB minimum (65,536 512-byte blocks)
rss	Maximum amount of physical memory a that user's process can use	64 MB minimum (131,072 512-byte blocks) or more

AIX User (shell) parameters

Table 4-9 lists the additional per-user (shell) parameters on AIX.

Table 4-9 AIX user (shell) parameters

User parameter	Engine tier	Service tier
MAXUPROC	200	1,000 or unlimited
NOFILES	1,000	10,000

In addition, on AIX platforms, tuning the additional kernel parameters listed in Table 4-10 might improve performance for the InfoSphere Information Server parallel engine. Work with your AIX system administrator to determine optimal values.

Table 4-10 Additional kernel parameters

User parameter	Description	Suggestion
maxperm	Maximum number of permanent buffer pages for file I/O	Tune to limit the amount of physical memory used for file system I/O so that more memory is dedicated for Enterprise Edition (EE) processes.
somaxconn	Maximum number of socket connections	Might need to increase this value for large EE jobs with lots of processes. When increasing maxuproc is not enough, increase this kernel parameter.
rbr	Release-behind-when-reading	Consider setting this file system mount option for permanent data set storage. Because large permanent data sets are always read sequentially and never in reverse or random order, this might improve performance.

User parameter	Description	Suggestion
CIO	Concurrent I/O	Investigate the use of CIO for both disk and scratch disk resources. CIO results in a performance similar to raw devices, which might benefit access to both permanent data sets and temporary sort files. CIO is good for I/O with large block sizes.

4.12.4 HP-UX system configuration

This section provides the HP-UX kernel parameters.

HP-UX kernel parameters

Table 4-11 lists the minimum requirements for UNIX kernel parameters on HP-UX installations of IBM InfoSphere Information Server.

After installing DB2 on HP-UX, run the **db2osconf** command to verify the suggested kernel settings.

Attention: For readability, the table values in this section include commas. Omit them when setting parameters.

Table 4-11 HP-UX kernel parameters

Kernel parameter	Description	InfoSphere Information Server minimum	WebSphere Application Server minimum	Notes
MAXFILES	Maximum number of open files per process		1448	
MSGMAP	Number of entries in SystemV IPC message space resource map		2319	
MSGMAX	Maximum message size in bytes	32,768	65,535	
MSGMNB	Maximum bytes per message queue	32,768	65,535	
MSGMNI	Maximum number of system-wide SystemV IPC message queues		2317	

Kernel parameter	Description	InfoSphere Information Server minimum	WebSphere Application Server minimum	Notes
MSGSEG	Number of system VIPC message segments	7,168	32,767	
MSGSZ	Message size		32	
MSGTQL	Maximum number of SystemV IPC messages		2317	
NFILE	Maximum number of files open simultaneously		16,219	
NFLOCKS	Maximum number of file locks		5793	
NIDNOE	Maximum number of HFS file system open inodes		4055	
NPROC	Number of simultaneous processes		2912	
SEMMNI	Number of semaphore identifiers (system wide)		2896	
SEMMNS	Total number of semaphores		5794	
SEMMNU	Maximum number of undo structures		2896	
SHMMAX	Maximum shared memory segment size	307,200,000	2,897,215,488	If disk caching is turned on and DISKCACHE is larger than 512, this must be set higher.
SHMSEG	Maximum number of shared memory segments per process	200		

On 32-bit platforms, HP-UX limits the maximum size of a shared memory segment to 1.75 GB. When processing extremely large in-memory reference tables in a DataStage Lookup Stage, you might need to configure HP-UX memory windowing. For details about this process, see “Example user setup for UNIX environments” on page 164.

HP-UX user (shell) parameters

Table 4-12 lists the additional per-user (shell) parameters that should be adjusted on HP-UX to ensure that the InfoSphere Information Server parallel engine has sufficient resources.

Table 4-12 HP-UX user (shell) parameters

User parameter	Description	Suggested setting
maxdsiz	Maximum size of data segment	2,039,480,320 (1945 MB)
maxssiz	Maximum size of stack	82,837,504 (79 MB)
maxtsiz	Maximum size of text segment	1,073,741,824
rss	Maximum amount of physical memory that a user's process can use	64 MB minimum (131,072 512-byte blocks) or more

4.12.5 RedHat and SUSE Linux system configuration

You cannot install InfoSphere Information Server on a version of RedHat Linux 4 that has been upgraded from RedHat v3 (upgrade configurations are missing required libraries).

RedHat and SUSE Linux kernel parameters

Table 4-13 lists the minimum requirements for UNIX kernel parameters on RedHat Linux installations of IBM InfoSphere Information Server.

Attention: For readability, the table values in this section include commas. Omit them when setting parameters.

Table 4-13 RedHat and SUSE Linux kernel parameters

Kernel parameter	Description	Engine tier	DB2 minimum	Services tier	Notes
MSGMAX	Maximum message size in bytes	8,192	65,536	No requirement	
MSGMNB	Maximum bytes per message queue	16,384	65,536	No requirement	

Kernel parameter	Description	Engine tier	DB2 minimum	Services tier	Notes
MSGMNI	Maximum queue system wide	No requirement	1024	No requirement	
SEMMNI	Number of semaphore identifiers (system wide)	1024	1024	No requirement	
SEMMNS	Total number of semaphores	128,000	256,000	No requirement	
SEMMSL	Maximum number of semaphores per id list	250	250	No requirement	
SEMOPM	Number of operations per semop call	32	32	No requirement	
SHMALL	Maximum total shared Memory		838608 KB	2511724800	
SHMMAX	Maximum shared memory segment size	307,200,000	32,768 KB	2511724800	If disk caching is turned on and DISKCACHE is larger than 512, this must be set higher.
SHMMNI	Shared memory identifiers	2000	4096	No requirement	
SHMSEG	Maximum number of shared memory segments per process	200			
Rlim_fd_max		No requirement		>=8193	
Rlim_fd_cur		No requirement		.+8193	
MAXUPROC		200		1,000 or unlimited	
NOFILES		1,000		10,000	

4.12.6 Solaris 9 system configuration

This section describes the Solaris kernel parameters.

Solaris kernel parameters

Table 4-14 lists the minimum requirements for UNIX kernel parameters for Solaris 9 installations of IBM InfoSphere Information Server.

After installing DB2 on Solaris 9, run the **db2osconf** command to verify the suggested kernel settings.

Attention: For readability, the table values in this section include commas. Omit them when setting parameters.

Table 4-14 Solaris 9 kernel settings

Kernel parameters	Description	InfoSphere Information Server minimum	WebSphere Application Server minimum	Notes
MSGMAP	Number of entries in message map		1026	
MSGMAX	Maximum message size in bytes	8,192	65,535	
MSGMNB	Maximum bytes per message queue	16,384		
MSGMNI	Maximum message queues (system wide)	1024		
SEMAEM	Maximum adjust-on-exit		16,384	
SEMMAP	Number of entries in semaphore map		1024	
SEMMNI	Number of semaphore identifiers (system wide)	1024		
SEMMNS	Total number of semaphores	128,000	16,384	
SEMMNU	Maximum number of undo structures		2048	
SEMMSL	Maximum number of semaphores per ID list	1024	100	

Kernel parameters	Description	InfoSphere Information Server minimum	WebSphere Application Server minimum	Notes
SEMOPM	Maximum number of semaphore operations	32	100	
SEMUME	Maximum number of undo structures per process		256	
SHMMAX	Maximum shared memory segment size	307,200,000	4,294967,295	If disk caching is turned on and DISKCACHE is larger than 512, this must be set higher.
SHMMNI	Shared memory identifiers	2000		
SHMSEG	Maximum number of shared memory segments per process	1024		

4.12.7 Solaris 10 system configuration

InfoSphere Information Server has not been certified to run in Solaris 10 zones or projects. If you choose to deploy InfoSphere Information Server on a Solaris zone or project, any issues raised with IBM Support must first be reproduced in a native operating system environment.

Solaris 10 kernel parameters

Table 4-15 on page 54 lists minimum requirements for UNIX resource parameters for Solaris 10 installations of IBM InfoSphere Information Server. Starting with Solaris 10, many older kernel parameters have been superseded by resource limits.

After installing DB2 on Solaris 10, run the **`db2osconf`** command to verify the suggested kernel settings.

Attention: For readability, the table values in this section include commas. Omit them when setting parameters.

Table 4-15 Solaris 10 kernel settings

Kernel parameters	Description	InfoSphere Information Server minimum	WebSphere Application Server minimum	Notes
MSGMAX	Maximummessage size in bytes	8,192	65,535	
MSGMNB	Maximum bytes per message queue	16,384		
SEMMNI	Number of semaphore identifiers (system wide)	1024		
SEMMSL	Maximum number of semaphores per id list	1024	100	
SEMOPM	Maximum number of semaphore operations	1024	100	
SHMMAX	Maximum shared memory segment size	6 GB	5	If disk caching is turned on and DISKCACHE is larger than 512, this value must be set higher.
SHMMNI	Shared memory identifiers	2000		

4.13 Verifying connectivity and network configuration

IBM InfoSphere Information Server relies on TCP/IP network infrastructure for communication between:

- InfoSphere Information Server clients and InfoSphere Information Server domain
- InfoSphere Information Server clients and Information Sever engine
- InfoSphere Information Server domain and metadata repository
- InfoSphere Information Server domain and Information Analyzer Repository
- InfoSphere Information Server engine and source/target data stores:
  - Databases
  - File transfer through FTP or named pipes
  - WebSphere MQ
  - Third-party applications such as SAS, Siebel, PeopleSoft, SAP

- Across InfoSphere Information Server engines in a clustered or Grid configuration
- InfoSphere Information Server engine and domain Server
- InfoSphere Information Server Job Monitor and InfoSphere Information Server engines
- InfoSphere Information Server Performance Monitor and InfoSphere Information Server engines

InfoSphere Information Server requires access to a number of TCP ports, which must be open, if necessary, through firewall configuration. Related connectivity (for example, databases and enterprise applications) might require additional port assignments. Consult vendor-supplied documentation for their requirements.

For a complete list of network ports used by InfoSphere Information Server, see "Network ports used by InfoSphere Information Server" on page 162.

InfoSphere Information Server network requirements

InfoSphere Information Server has the following network requirements:

- TCP/IP network is required.
- All servers must be resolvable by name, between tiers and from clients to servers.
- For optimal performance, all layers of InfoSphere Information Server should be installed on the same high-speed, low-latency local area network.
- In a cluster or grid deployment, all engine servers must be located in the same physical data center, and should be connected by a dedicated, private high-speed network connection.
- The domain and metadata repository database should be installed on the same server. If they are installed on separate servers, the domain and metadata repository database should be connected by a dedicated, private, high-speed network connection.
- Due to the data exchanges that occur between the InfoSphere Information Server layers, it is strongly suggested that all tiers be located in the local area network (LAN). When deploying in a WAN configuration, use a network hosting tool (for example, Citrix or Windows Remote Desktop) to host the InfoSphere Information Server clients.

4.14 Configuring operating system users, groups, and permissions

This section highlights system configurations.

4.14.1 Privileged installation user

IBM InfoSphere Information Server requires a privileged user account for installation.

- On UNIX platforms, the installation must be performed by root or by a user account with root privileges.
- On Windows environments, the installation must be run from a local administrator account. This user must have read/write access to the target installation directories. The installation cannot be run from a domain administrator.

4.14.2 Required operating system users

InfoSphere Information Server requires a set of operating system user accounts to install the engine and metadata repository database, which are listed in Table 4-16. These accounts are used by the InfoSphere Information Server engine, and internal domain services.

Table 4-16 Operating system users

User account	Default user name	Primary group	Secondary group	Notes
DataStage administrator	dsadm	dstage		
DB2 administration server	dasusr1	dasadm1		Only needed for DB2.
DB2 instance owner	db2inst1	db2iadm1	dasadm1	Only needed for DB2.
DB2 fenced user	db2fenc1	db2fadm1		Only needed for DB2.
Metadata repository owner	xmeta	xmeta		DB2 uses OS authentication.
Information Analyzer analysis database owner	iauser	iauser		DB2 uses OS authentication.

Important: These users can be created by the installer, but this is not suggested.

The security configuration of many operating systems (for example, AIX) requires new users to log in before an account is activated.

For instructions to create users, see the *IBM Information Server Planning, Installation, and Configuration Guide*, GC19-1048-07. You can find a simple UNIX user setup in “Example user setup for UNIX environments” on page 164.

Operating system user requirements

Because of the way that the InfoSphere Information Server installer parses its parameters, passwords should not include embedded dollar signs ($).

4.14.3 Domain (WebSphere Application Server) user registry

IBM InfoSphere Information Server users log in and authenticate through the domain WebSphere Application Server. During install, two domain accounts must be specified, as listed in Table 4-17.

Table 4-17 Domain accounts

User account	Default user name	Notes
WebSphere administrator	wasadmin	
InfoSphere Information Server administrator	isadmin	Should be different from the WebSphere administrator

WebSphere Application Server user requirements

WebSphere Application Server has the following user requirements:

- During the InfoSphere Information Server installation, WebSphere Application Server can be configured to authenticate using an internal registry or using operating system users. This option can be changed later through the WebSphere Application Server Administration Console.
- When using OS authentication, user accounts must be created and activated before running the InfoSphere Information Server installation.
- WebSphere Application Server does not support the use of NIS as a user registry. The supported user registries are LDAP, OS, or an internal user registry.
- When installing in an LDAP environment, choose an internal user registry. LDAP authentication is configured after the InfoSphere Information Server installation.
- LDAP Servers supported by InfoSphere Information Server are those supported by WebSphere Application Server 6.0.2. A complete list of

supported LDAP Servers can be found in the list of WebSphere Application Server software requirements at:

http://www.ibm.com/support/docview.wss?rs=180&uid=swg27007256

- Because of the way that the InfoSphere Information Server installer parses its parameters, passwords should not include embedded dollar signs ($).

4.14.4 Engine (DataStage) user setup

The IBM InfoSphere Information Server engine must have at least one operating system user defined. DataStage and QualityStage jobs and Information Analyzer jobs run on the engine server using operating system user permissions.

- When using the WebSphere Application Server internal user registry, InfoSphere Information Server users must be mapped to at least one operating system user.
- When using the WebSphere Application Server OS or LDAP configuration, InfoSphere Information Server can be configured to share the user registry with the engine (DataStage/QualityStage) registry. This eliminates the need to individually map each InfoSphere Information Server user to an operating system or DataStage user.

DataStage supports four basic categories of users:

- Managers
- Developers
- Operators
- Super operators

These are implemented as InfoSphere Information Server roles that can be assigned to each user. The InfoSphere Information Server Console is used to assign either the DataStage Admin or DataStage User to each user. This allows the DataStage Administrator Client to assign each user to a particular role (operator, super operator, developer, and production manager) for a particular project.

Engine (DataStage) user setup on UNIX

DataStage can be administered on a UNIX platform by a special non-root user. This is *dsadm* by default, but you can specify a different administrative user at install. Set up this user before installing DataStage.

Each user is then allocated to the product manager, developer, operator, or super operator role (but not to more than one role per project). You can then use the DataStage Administrator to assign the appropriate DataStage user role to the user. For more information, see "User Roles on UNIX Systems" in the *IBM*

Information Server Administrator Guide, SC18-9929-02. Operators cannot use the DataStage Designer and only see released jobs in the DataStage Director. Neither operators nor developers can create protected projects or add anything to them.

More information: See "Setting Up Security" in the *IBM Information Server Administration Guide*, SC18-9929-02.

4.14.5 Engine (DataStage) user setup on Windows

On the Windows 2003 Server, DataStage must be installed by a user that has local administrator rights. This user must also have read/write permission to the target directory used to install the DataStage server.

If you are logged into a domain account, it must be part of the local administrators group on the server that you are installing, and you must have network access to the Windows domain controller for authentication.

4.15 Verifying and installing C++ compiler and runtime libraries

To develop parallel jobs with DataStage, install the C++ compiler that is specific to your platform on the same server as the InfoSphere Information Server engine. In development environments, DataStage uses the C++ compiler to generate parallel transformer and BuildOp components.

Many compilers are licensed on a per-user basis. DataStage only invokes the C++ compiler when the developer compiles a parallel job with transformer stages or compiles a BuildOp component. That is, the maximum number of simultaneous DataStage developers determines the maximum number of concurrent C++ compiler licenses.

On each platform, only one C++ compiler is supported. Check the InfoSphere Information Server system requirements for information about supported compilers and compiler installation requirements.

IBM InfoSphere Information Server system requirements

For InfoSphere Information Server system requirements, go to:

http://www.ibm.com/support/docview.wss?rs=14&uid=swg21315971#dqx1windowsdiskspace

For deployment (production) systems, a C++ compiler is not required. However, certain platforms might require C++ runtime libraries to be installed. This information is also contained with the InfoSphere Information Server system requirements.

4.16 Verifying InfoSphere Information Server connector requirements

InfoSphere Information Server uses connectors to access source and target data stores. Each connector (for example, ODBC, DB2, and Teradata) has specific requirements for setup and configuration that might have platform-specific steps. Consult the connector documentation and release notes for specific connector requirements.

4.17 Downloading and installing InfoSphere Information Server

This section explains the process for downloading and installing InfoSphere Information Server and required fix packs.

The installation packages for InfoSphere Information Server are available through IBM Passport Advantage. If you do not already have an account (login) for Passport Advantage, you will need to create one and make sure that it is associated with your customer ID to access all downloads for which you are licensed.

Get IBM Passport Advantage Software Downloads from:

http://www-306.ibm.com/software/howtobuy/passportadvantage/index.html

On certain platforms (specifically, 64-bit installations of RedHat Linux, SUSE Linux, and HP-UX) you might also need to download WebSphere Application Server and DB2 packages. You can find details about specific packages required by platform in the IBM InfoSphere Information Server system requirements at:

http://www-01.ibm.com/support/docview.wss?rs=14&uid=swg21315971#dqx1windowsdiskspace

- IBM InfoSphere Information Server 8.0.1 installer
- IBM InfoSphere Information Server 8.0.1 fix pack installer
- If applicable: WebSphere Application Server v6.0.2 installer
- If applicable: WebSphere Application Server fix pack installer

- If applicable: DB2 Enterprise Server Edition v9.1 installer
- If applicable: DB2 Enterprise Server Edition v9.1 fix pack installer

In addition, product-specific patches might be required for issues discovered after the latest fix pack release. Contact IBM Support for details on obtaining product patches.

4.18 Performing complete system backup

Before beginning an InfoSphere Information Server installation, it is important to have a complete system backup if the installation fails or you need to bring the system back to the state it was in before the installation began.

This backup should be performed by the system administrator on all target installation servers and should include both operating system and user directories.

4.19 Identifying and configuring file systems

DataStage requires file systems and space to be available for the following elements:

- Software Install Directory

  DataStage executables, libraries, and pre-built components
- DataStage Project (Repository) Directory
- Data Storage
  - DataStage temporary storage (scratch, temp, buffer)
  - DataStage parallel data set segment files
  - Staging and archival storage for any source files

By default, each of these directories (except for file staging) is created during installation as subdirectories under the base DataStage installation directory.

Important: Each storage class should be isolated in a separate file system to accommodate its different performance and capacity characteristics and backup requirements.

The default installation is generally acceptable for small prototype environments. Figure 4-4 illustrates how you might configure the file systems to satisfy the requirements of each class of DataStage storage.

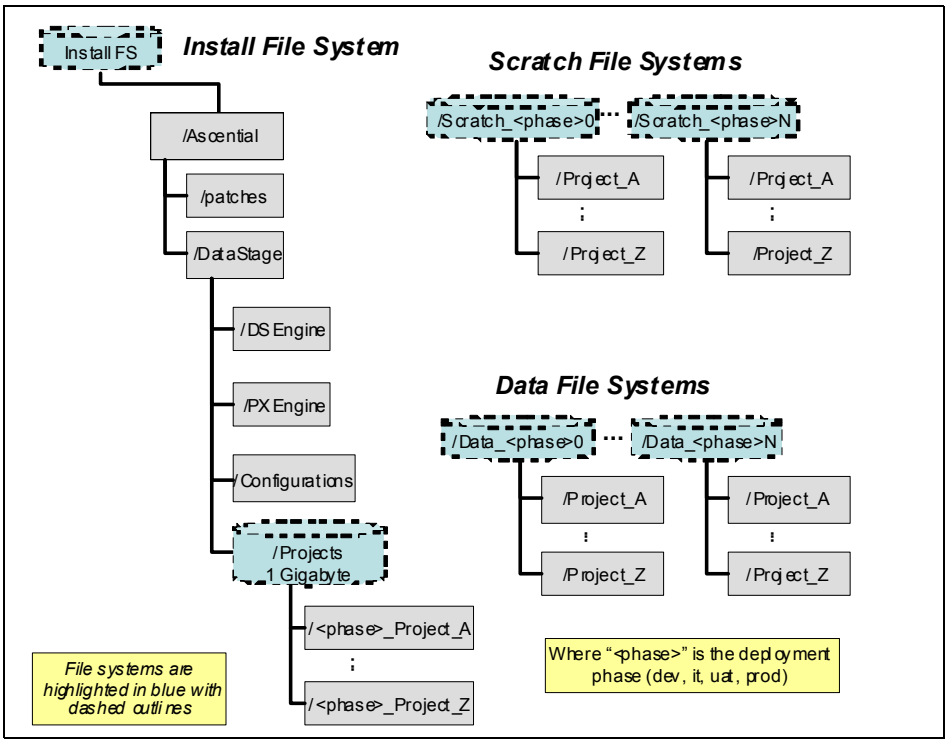


Figure 4-4 Suggested DataStage installation, projects, data, and scratch file systems

Notes about Figure 4-4:

- All file systems used by DataStage for installation must have read and write permissions for the primary DataStage users and groups.
- The DataStage installation directory should be reserved for installation of executables and libraries only. For cluster/grid implementations, it is generally best to share the installation mount point across servers.
- Notice that there is a projects file system under the DataStage directory. This allows optimal control of the DataStage project storage while still maintaining the default path used by the administrator client. Project naming standards include the deployment phase (dev, it, uat, prod) as a prefix. Ensuring that the phase is part of the project directory name will isolate projects in the same environment. For cluster/grid implementations, it is generally best to share the projects mount point across servers.

- Consider creating separate file systems for each scratch and data resource partition to scale DataStage I/O, naming the file systems in accordance with the partition numbers in your parallel configuration file. This standard practice advocates creating subdirectories for each project for each scratch and disk partition.
- File systems should be expandable without requiring destruction and recreation.
- In this document, the base install directory is referred to as `$DSROOT`, and we refer to the example installation directory as `/opt/IBM/InformationServer/`.

4.19.1 Software installation directory

The software installation directory is created by the installation process and contains the DataStage software file tree. The installation directory grows little over the life of a major software release. Therefore, the default location ($HOME for dsadm, for example, `/home/dsadm`) might be adequate.

The installation of InfoSphere Information Server requires the following minimum disk space for the listed environments:

- 1.3 GB for WebSphere Application Server
- 500 MB for DB2
- 1.4 GB for the InfoSphere Information Server components
- 2.5 GB for the metadata repository database
- 1 GB for the InfoSphere Information Analyzer analysis database
- 2 GB of temporary space during the installation

For cluster or grid implementations, it is generally best to share the installation file system across servers (at the same mount point).

Note: The DataStage installer attempts to rename the installation directory to support later upgrades. If you install directly to a mount point, this rename will fail, and several error messages will be displayed. The installation will succeed but the messages might be confusing.

4.19.2 DataStage Projects (repository) directory

The DataStage Projects subdirectory contains the repository (universe database files) of job designs, design and runtime metadata, logs, and components. Project directories can grow to contain thousands of files and subdirectories depending on the number of projects, the number of jobs, and the volume of logging information retained about each job.

During the installation process, the projects subdirectory is created in the DataStage install directory. By default, the DataStage Administrator client creates its projects in this projects subdirectory.

For cluster or grid implementations, it is generally best to share the projects file system across servers (at the same mount point).

Important: It is not a good practice to create DataStage projects in the default directory within the install file system, as disk space is typically limited. Projects should be created in their own file system (Figure 4-5).

Creating the projects file system

On most operating systems, you can create separate file systems at non-root levels, which is illustrated in Figure 4-5 on page 67, as a separate file system for the projects subdirectory within the DataStage installation. Use the following guidelines:

- Create a separate file system and mount it over the default location for projects, the `$DSROOT/Projects` directory. Mount this directory after installing DataStage but before projects are created.
- The projects directory should be a mirrored file system with sufficient space (minimum 100 MB per project).
- For cluster or grid implementations, share the project file system across servers (at the same mount point).

Important: Monitor the project file system to ensure that adequate free space remains. If the project file system runs out of free space during DataStage activity, the repository might become corrupted, requiring a restore from backup.

Effective management of space is important to the health and performance of a project, and as jobs are added to a project, new directories are created in this file tree, and as jobs are run, their log entries multiply. These activities cause file system stress. For example, they result in more time to insert or delete DataStage components and longer update times for logs. Failure to perform routine project maintenance (for example, remove obsolete jobs and manage log entries) can cause project obesity and performance issues.

The name of a DataStage Project is limited to a maximum of 54 characters. The project name can contain alphanumeric characters and underscores (_).

Maintain project names in unison with source code control. As projects are promoted through source control, the name of the phase and the project name should reflect the version, in the form:

<Phase>_<ProjectName>_<version>

Where *Phase* corresponds to the phase in the application development life cycle (Table 4-18).

Table 4-18 Development life cycle

Phase name	Phase description
Dev	Development
IT	Integration test
UAT	User acceptance test
Prod	Production

Project recovery considerations

Devising a backup scheme for project directories is based on the following three core issues:

- Will there be valuable data stored in Server Edition hash files?

  Note: The use of Server Edition components in an Enterprise Edition environment is discouraged for performance and maintenance reasons. However, if Server Edition applications exist, their corresponding objects might need to be taken into consideration.

  DataStage Server Edition files located in the DataStage file tree might require archiving from a data perspective.

- How often will the UNIX file system containing the entire DataStage file tree be backed up? When can DataStage be shut down to enable a cold snapshot of the universe database and the project files? A complete file system backup while DataStage is shut down accomplishes this backup.
- How often will the projects be backed up? Keep in mind that the grain of project backups will represent the ability to recover lost work should a project or a job become corrupted.

At a minimum, a UNIX file system backup of the entire DataStage file tree should be performed at least weekly with the DataStage engine shut down, and each project should be backed up with the manager at least nightly with all users

logged out of DataStage. This is the equivalent of a cold database backup and six updates.

If your installation has valuable information in server hash files, increase the frequency of your UNIX backup *or* write jobs to unload the server files to external media.

4.19.3 Data set and sort directories

The DataStage installer creates the following two subdirectories within the DataStage installation directory:

- The `Datasets/` subdirectory stores individual segment files of DataStage parallel data sets.
- The `Scratch/` subdirectory is used by the DataStage framework for temporary files for such things as sort and buffer overflow.

Try not to use these directories, and consider deleting them to ensure that they are never used. This is best done immediately after installation, but be sure to coordinate this standard with the rest of the team.

DataStage parallel configuration files are used to assign resources (such as processing nodes, disk, and scratch file systems) at run time when a job is executed. Parallel configuration files are discussed in detail in Chapter 5, “Parallel configuration files” on page 123.

The DataStage installer creates a default parallel configuration file (`configurations/default.apt`) that references the datasets and scratch subdirectories within the install directory. The DataStage administrator should consider removing the `default.apt` file altogether, or at a minimum updating this file to reference the file systems that you define.

Data and scratch file systems

It is not a good practice to share the DataStage install and projects file systems with volatile files such as scratch files and parallel data set segment files. Resource, scratch, and sort disks service different kinds of data with opposite persistence characteristics. Furthermore, they compete directly with each other for I/O bandwidth and service time if they share the same path.

Optimally, these file systems should not have any physical disks in common and should not share any physical disks with other applications. While it is often impossible or impractical to allocate contention-free storage, it must be noted that at large data volumes and in highly active job environments, disk arm contention can and usually does significantly constrain performance.

For best performance, and to minimize storage impact on development activities, create separate file systems for each data and scratch resource partition. This practice advocates creating subdirectories for each project within each scratch and disk partition (Figure 4-5).

On clustered and grid environments, data and scratch file systems should not be shared across servers. Each server contains its own subset of the data.

On systems where multiple phases are shared on the same server, consider separating data and scratch storage to different file systems for each deployment phase to completely isolate each environment. Figure 4-5 illustrates this (where <*phase*> is dev, it, uat, or prod). Use of the phase name is not required when environments are not shared on the same system.

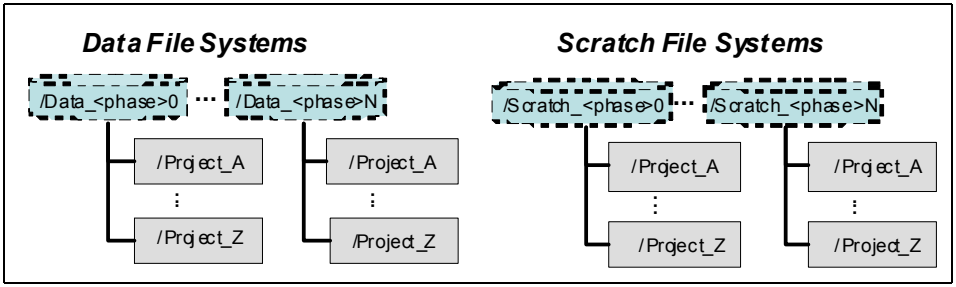


Figure 4-5 Suggested data and scratch file systems

Note: For optimal performance, create file systems in high-performance, low-contention storage. The file systems should be expandable without requiring destruction and recreation.

Data sets

Parallel data sets are used for persistent data storage in parallel, in native DataStage format. The DataStage developer specifies the location of the data set header file, which is a small pointer to the actual data segment files that are created by the DataStage engine, in the directories specified by the disk resources assigned to each node in the parallel configuration file. Over time, the data set segment file directories will grow to contain dozens to thousands of files, depending on the number of DataStage data sets used by DataStage jobs.

The need to archive data set segment files depends on the recovery strategy chosen by the DataStage developer, the ability to recreate these files if the data sources remain, and the business requirements. Whatever archive policy is chosen should be coordinated with the DataStage administrator and developers.

If data set segment files are archived, careful attention should be made to also archive the corresponding data set header files.

Sort space

As a suggested practice, isolate DataStage scratch space from data sets and flat files, and DataStage sort space, because temporary files exist only while a job is running, and they are warm files (that is, they are being read and written at above average rates). Certain files created by database stages persist after job completion. For example, the Oracle `.log`, `.ctl` and `.bad` files will remain in the first scratch resource pool after a load completes.

Note: The sort space must accommodate only the files being sorted concurrently, and, assuming that jobs are scheduled non-concurrently, only the maximum of those sorts. There is no persistence to these temporary sort files, so they do not need to be archived.

Sizing DataStage scratch space is somewhat difficult. Objects in this space include lookups and intra-process buffers. Intra-process buffers absorb rows at run time when stages in a partition (or all partitions) cannot process rows as fast as they are supplied. In general, there are as many buffers as there are stages on the canvas for each partition. As a practical matter, assume that scratch space must accommodate the largest volume of data in one job. There are advanced ways to isolate buffer storage from sort storage, but this is a performance-tuning exercise, not a general requirement.

Maintaining parallel configuration files

DataStage parallel configuration files are used to assign resources (such as processing nodes, disk, and scratch file systems) at run time when a job is executed. For more information about parallel configuration files, see Chapter 5, “Parallel configuration files” on page 123.

Parallel configuration files can be located within any directory that has suitable access permissions, defined at run time through the environment variable $APT_CONFIG_FILE. However, the graphical configurations tool within the DataStage clients expects these files to be stored within the configurations subdirectory of the DataStage install. For this reason, it is suggested that all parallel configuration files be stored in the configurations subdirectory, with naming conventions to associate them with a particular project or application.

The `default.apt` file is created when DataStage is installed, and references the datasets and scratch subdirectories of the DataStage install directory. To manage system resources and disk allocation, the DataStage administrator

should consider removing this file, creating separate configuration files that are referenced by the $APT_CONFIG_FILE setting in each DataStage project.

At a minimum, the DataStage administrator should edit the `default.apt` configuration file to reference the newly created data and scratch file systems and to ensure that these directories are used by any other parallel configuration files.

4.19.4 Extending the DataStage project for external entities

Create another directory structure, referred to as `Project_Plus`, to integrate all aspects of a DataStage application that are managed outside of the DataStage Projects repository. The `Project_Plus` hierarchy includes directories for secured parameter files, data set header files, custom components, Orchestrate schema, SQL, and shell scripts. It might also be useful to support custom job logs and reports.

The `Project_Plus` directories provide a complete and separate structure in the same spirit as a DataStage project, organizing external entities in a structure that is associated with one and only one corresponding DataStage project. This provides a convenient vehicle to group and manage resources used by a project.

It is common for a DataStage application to be integrated with external entities, such as the operating system, enterprise schedulers and monitors, resource managers, other applications, and middle ware. The `Project_Plus` directory provides an extensible model that can support this integration through directories for storing source files, scripts, and other components.

Project_Plus and change management

Project naming conventions recommend naming a project with a prefix to indicate the deployment phase (dev, it, uat, prod). Following this naming convention will also separate the associated files within the corresponding `Project_Plus` hierarchy.

However, to completely isolate support files in a manner that is easy to assign to separate file systems, an additional level of directory structure can be used to enable multiple phases of application deployment (development, integration testing, user acceptance testing, and production) as appropriate. If the file system is not shared across multiple servers, not all of these development phases might be present on a local file system.

Project_Plus file system

The `Project_Plus` directory is often stored in the `/usr/local` home directory (for example, `/usr/local/dstage`), but this can be in any location as long as permissions and file system access are permitted to the DataStage developers and applications.

Note: The file system where the Project_Plus hierarchy is stored must be expandable without requiring destruction and recreation.

Project_Plus directory structure

Figure 4-6 illustrates typical components and structure of the `Project_Plus` directory hierarchy.

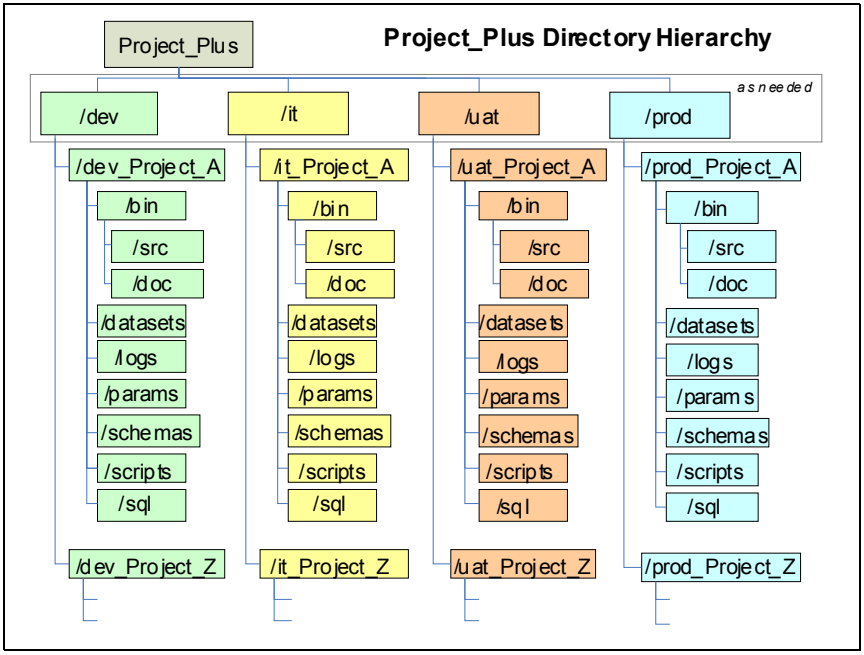


Figure 4-6 Project_Plus directory structure

Table 4-19 provides a description of the Project_Plus directory items.

Table 4-19 Project_Plus directory descriptions

Directory	Description
Project_Plus	Top-level of directory hierarchy.
/dev	Development phase directory tree (if applicable).
/dev_Project_A	Subdirectory created for each DataStage project (the actual directory name dev_Project_A should match the corresponding DataStage project name).
/bin	Location of custom programs, DataStage routines, BuildOps, utilities, and shells.
/doc	Documentation for programs in /bin subdirectory.
/src	Source code and Makefiles for items in /bin subdirectory. Note: Depending on change management policies, this directory might only be present in the /dev development phase directory tree.
/datasets	Location of parallel data set header files (.ds files).
/logs	Location of custom job logs and reports.
/params	Location of parameter files for automated program control, a backup copy of dsenv, and backup copies of DSParams:$ProjectName project files.
/schemas	Location of Orchestrate schema files.
/it	Integration test phase directory tree (if applicable).
/uat	User acceptance test phase directory tree (if applicable).
/prod	Production phase directory tree (if applicable).

Project_Plus environment variables

The Project_Plus directory structure is made to be transparent to the DataStage application, through the use of environment variable parameters used by the DataStage job developer. Environment variables are a critical portability tool, which enable DataStage applications to be deployed through the life cycle without any code changes.

In support of a `Project_Plus` directory structure, configure the user-defined environment variable parameters (Table 4-20) for each project using the DataStage Administrator, substituting your `Project_Plus` file system and project name in the value column:

Table 4-20 User-defined environment variables

Name	Type	Prompt	Value
PROJECT_PLUS_DATASETS	String	Project + dataset descriptor dir	Project_Plus/devProject_A/datasets/
PROJECT_PLUS_LOGS	String	Project + log dir	/Project_Plus/devProject_A/logs/
PROJECT_PLUS_PARAMS	String	Project + parameter file dir	Project_Plus/devProject_A/params/
PROJECT_PLUS_SCHEMAS	String	Project + schema dir	Project_Plus/devProject_A/schemas/
PROJECT_PLUS_SCRIPTS	String	Project + scripts dir	/Project_Plus/devProject_A/scripts/

Note: The `Project_Plus` default values include a trailing directory separator to avoid having to specify in the stage properties. This is optional, but whatever standard the administrator chooses, it should be set and consistently deployed across projects and job designs.

Figure 4-7 depicts the `Project_Plus` environment variables.

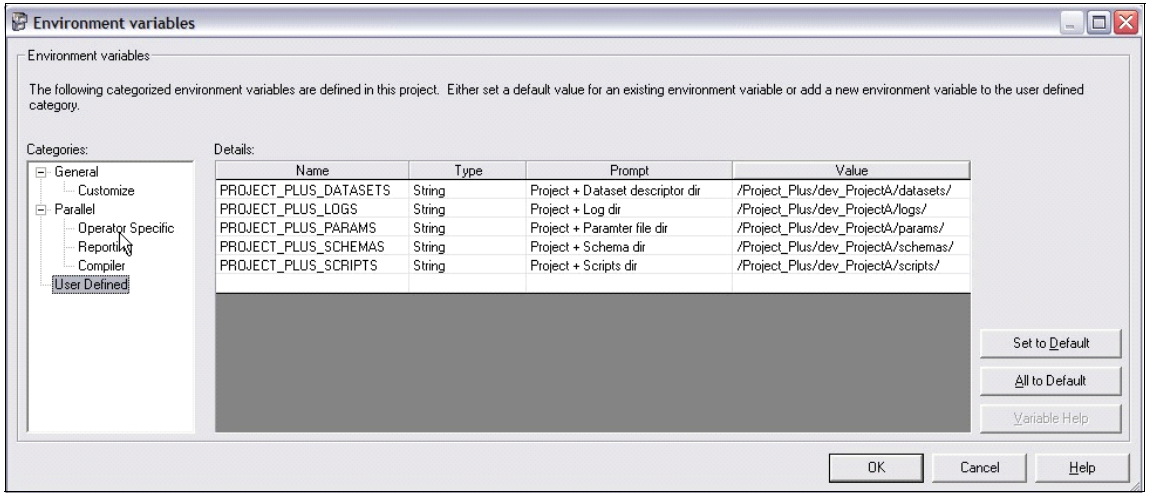


Figure 4-7 Project_Plus environment variables

Using Project_Plus with grid or cluster deployments

When deploying a DataStage application in cluster or grid environments, or when configuring for high-availability and disaster recovery scenarios, careful consideration should be made when sharing the `Project_Plus` file system configuration.

In general, the custom components, data set header files, and other components of the `Project_Plus` directory should be visible to all members of the cluster or grid, using the same mount point on all servers. Creation of small individual mounts points is generally not desirable.

Mount this directory on all members of the cluster after installing DataStage, but before creating any DataSets.

4.19.5 File staging

Use a separate staging file system and directory structure to store, manage, and archive various source data files (Figure 4-8).

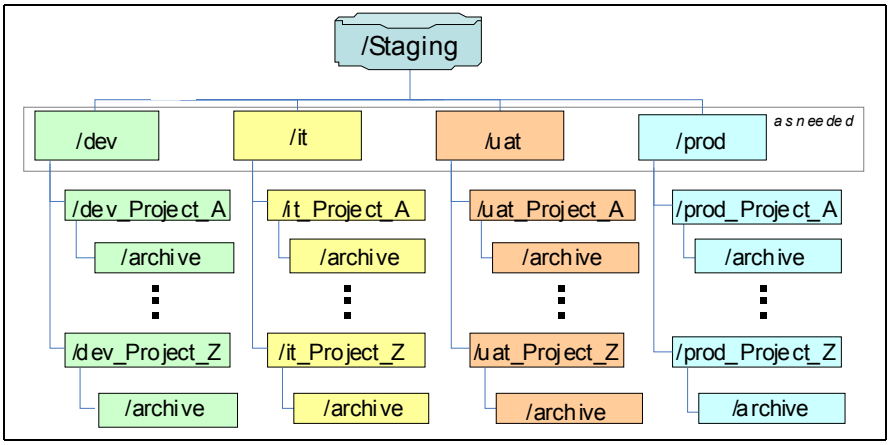


Figure 4-8 DataStage staging directories

Project naming conventions recommend naming a project with a suffix to indicate the deployment phase (dev, it, uat, prod). Following this naming convention will also separate the associated files within the corresponding staging hierarchy.

However, to completely isolate support files in a manner that is easy to assign to separate file systems, an additional level of directory structure can be used to enable multiple phases of application deployment (development, integration test, user acceptance test, and production), as appropriate. If the file system is not shared across multiple servers, not all of these development phases might be present on a local file system.

In support of the staging directory structure, the user-defined environment variable parameters (Table 4-21) should be configured for each project using the DataStage Administrator, substituting your staging file system and project name in the value column.

Table 4-21 User-defined environment variable parameters

Name	Type	Prompt	Example value
STAGING_DIR	String	Staging directory	/Staging/
PROJECT_NAME	String	Project name	devProject_A
DEPLOY_PHASE	String	Deployment phase	dev

The Project_Name and Deploy_Phase variables are used to properly parameterize the directory location within job designs.

Note: The STAGING_DIR default value includes a trailing directory separator to avoid having to specify in the stage properties. This is optional, but whatever standard the administrator chooses, it should be set and consistently deployed across projects and job designs.

Within each deployment directory, files are separated by project name (Table 4-22).

Table 4-22 Deployment directory files

Directory	Description
`Staging`	Top-level of directory hierarchy
`/dev`	Development phase directory tree (if applicable)
`/dev_Project_A`	Subdirectory created for each DataStage project (The actual directory name `dev_Project_A` should match the corresponding DataStage Project Name.) Location of source data files, target data files, and error and reject files
`/archive`	Location of compressed archives created by archive process of previously processed files
`/it`	Integration test phase directory tree (if applicable)
`/uat`	User acceptance test phase directory tree (if applicable)
`/prod`	Production phase directory tree (if applicable)

4.19.6 File system sizing example

File systems should be expandable to accommodate growth from prototype to full-scale development to test and deployment (as applicable) without requiring destruction and recreation.

Accurate capacity planning requires detailed requirements for persistence, failover and archive, growth volumes, and projections across all applications to be deployed in the environment.

An effective method of specifying the storage requirements is a table such as Table 4-23 on page 76. In this example, two logical processing nodes are

configured to process 100 GB of raw data. Our space calculation is per logical node, so for data, the formula would be 100 x 2.5 / 2 or 125 GB, assuming that the data is fairly evenly distributed between nodes. We describe our requirements in terms of contention objectives wherever possible.

Table 4-23 Storage requirements

File system type	Size (GB)	Mount point	Use and contention requirement
Data set	125	`/RaidVol1/Data0`	One-half the volume of DataStage DataSets. This file system should be expandable without removal and recreation, should not share physical disks with other DataSets or database files, should be highly available, and should have as many physical disks as possible, with a preferred minimum of 4.
Data set	125	`/RaidVol1/Data1`	Same as above.
Scratch	125	`/RaidVol1/Scratch0`	One-half the volume of DataStage scratch (working) space. This file system should be expandable without removal and recreation, should not share physical disks with other DataSets or database files, should be highly available, and should have as many physical disks as possible, with a preferred minimum of 4.
Scratch	125	`/RaidVol1/Scratch1`	Same as above.
Install	2	`/usr/dsadm`	DataStage software. This space must be highly available.
Projects	2	`/usr/dsadm/Ascential/DataStage/Projects`	
Project_Plus	1	`/export/dsadm`	DataStage utilities and shared files. This space must be highly available.
Staging	125	`/datavol1`	Source data. This file system should be expandable without removal and re-creation, should be highly available, and should have as many physical disks as possible, with a preferred minimum of 4.

4.20 Connectivity and network configuration

DataStage relies on network infrastructure for communication between:

- DataStage clients and DataStage server
- DataStage server nodes and source/target data stores (for example, databases, file through FTP or named pipes, third-party applications such as SAS, Siebel, PeopleSoft, and SAP)
- Across DataStage server nodes in a clustered or grid configuration

For optimal performance, each of these network connections should be on high-speed connections with low latency, particularly the server-to-server network connections for clustered/grid configurations and access to remote sources/targets.

4.20.1 Network port usage

DataStage requires access to a number of TCP ports, which must be opened (if necessary, through firewall configuration). Related connectivity (for example, databases and enterprise applications) might require additional port assignments. Consult vendor-supplied documentation for their requirements. Further details on firewall configuration can be found in “Windows XP Service Pack 2 firewall configuration” on page 147 and “Network ports used by InfoSphere Information Server” on page 162.

Table 4-24 lists ports with descriptions.

Table 4-24 Ports and descriptions

Port	Use	Description or notes
31538	DSRPC daemon (client/server) connectivity	Default for v7.0 and later.
>= 10,000	Conductor/section leader messages	At job startup, tries to allocate port starting from $APT_PM_STARTUP_PORT (default 10,000) until an unused port is located.
>= 11,000	Player - player data transfer	On cluster/grid configurations, tries to allocate a range of ports starting at $APT_PLAYER_CONNECTION_PORT (default 11,000) based on unused ports. Number of ports required varies based on job complexity (number of stages) and degree of parallelism.
2000	RTI Agent	Static port assignment (default).

Port	Use	Description or notes
>1024 assigned by OS	RTI Agent	Dynamic Port Assignments. For example, RTI Agent opens a port to allow notification messages to be sent from DataStage Jobs. When an RTI Server sends data to the RTI Agent, both have additional ports open to handle that exchange.
1099 and 8083	RTI - Jboss Application Server	EJB Server.
1476	RTI - Jboss Application Server	Internal Database Server.
8090 and 8091	RTI - Jboss Application Server	JMS Services.
4444	RTI - Jboss Application Server	JMX Service.
8080	RTI - Jboss Application Server	HTTP.
2379	MetaStage - Listener	Default (can be configured).
4379	MetaStage - Process Metabroker	Default (can be configured).
9090	MetaStage - Browser	Default (can be configured).

Network port usage on UNIX System Services platforms

On UNIX System Services platforms, if you are using an rsh connection (a tightly coupled environment) and want to collect process metadata via the job monitor, you will need to assign two TCP ports to DataStage.

The default numbers for these ports are 13400 and 13401. The numbers are specified at installation time and are easily modified afterwards.

4.20.2 UNIX NIS configuration

If you are using NIS, the master `/etc/services` file must be updated manually to add the following entry:

```
dsrpc 31538/tcp
```

The local `/etc/passwd` file must be updated to include a dummy entry for the DataStage administration user (for example, dsadm) for the installation to work.

4.20.3 Windows network configuration

On Windows 2003 Server environments, the default port assignments for the Microsoft Telnet Server are the same as those used by the DataStage Telnet server and the MKS Telnet server. This can cause the DataStage install to hang when starting its services.

On Windows 2003 systems, you should reconfigure these services so that the ports do not overlap. Or you can selectively disable startup of the conflicting services:

- MKS Telnet Server is not needed or used by DataStage.
- DataStage Telnet Server is only used for remote debugging of the internal DataStage repository engine, and is not necessary for normal operation.

4.21 Configuring OS users, groups, and associated permissions

Before beginning a DataStage installation, operating system groups and users must be selected or implemented. DataStage supports three basic categories of users:

- Managers
- Developers
- Operators

These are implemented as the secondary group for each category of user. That is, the primary group ID (GID) of every DataStage user must be the same as the DataStage super-user, and the secondary GID of a user in one of these categories must be one of these groups. The DataStage Client Administrator function is used to assign roles to those users who are members of each group.

Security is best supported when there are four operating system groups associated with DataStage (Table 4-25).

Table 4-25 DataStage groups

Group	Function
dstage	The primary group of the DataStage super-user. By convention, this user is dsadm.
dsdev	The secondary group of DataStage developers. No users in this group are members of any other DataStage group.

Group	Function
dsmgr	The secondary group of DataStage managers. No users in this group are members of any other DataStage group.
dsopr	The secondary group of DataStage operators. Typically, this group is the primary group for schedule management software user IDs, for example, asys. No users in this group are members of any other DataStage group.

Minimally, the only group that must be created is the primary group of the DataStage super-user. This group and the super-user (dsadm) are used to secure files and perform software maintenance.

Note: The DataStage administrator must have unrestricted access to the super-user (dsadm).

An effective method of specifying the groups and users is a table, such as the one depicted in Table 4-26. On our example system, users and groups in bold already exist. The remaining groups and users are to be created. We describe our requirements in terms of business functions wherever possible and identify employees, contractors, and virtual users, such as dsadm.

Table 4-26 Groups and users

Primary group	Secondary group	User	Role
dstage	dsmgr	dsadm	DataStage software and project manager. This user ID manages the software and project population and must exist on all systems (development, test, and production systems). The password will be closely controlled.
	dsmgr	johnl	DataStage project manager. This user is an employee who manages the development and testing project environments and is responsible for project backup and recovery, job migration to and from development, source control, and production, and must exist on both development and production systems.
	dsmgr	paulm	Paul is an employee who is John's back up. He has the same roles as John, and must have access to both production and development systems.
dstage	dsopr, autosys	asys	This user ID runs the jobs in the system integration and production environments. Autosys is a popular job scheduler for UNIX and is used as an example.

Primary group	Secondary group	User	Role
dstage	dsdev, users	georgeh	DataStage developer. This user is an employee who needs access to the development and test systems.
	dsdev, users	richards	DataStage developer. This user is an employee who needs access to the development and test systems.
	dsdev, users	mannie	DataStage developer. This user is a contractor who needs access to the development and test systems.
	dsdev, users	moe	DataStage developer. This user is a contractor who needs access to the development and test systems.
	dsdev, users	jack	DataStage developer. This user is a contractor who needs access to the development and test systems.

4.21.1 UNIX user configuration

DataStage can be administered on a UNIX platform by a special non-root user, which is dsadm by default. However, you can specify a different administrative user at installation. You must set up this user before installing DataStage. All DataStage users should belong to the same UNIX group, and this should be the administrative user's primary group. We suggest that you name the group *dstage*.

If you want to set up the system so that it distinguishes between product managers, developers, and operators, set up secondary groups for each class of user. Each user is then allocated to the product manager, developer, or operator secondary group (but not to more than one secondary group). You can then use the DataStage administrator to assign the appropriate DataStage user role to the secondary groups. For more information, see "User Roles on UNIX Systems" in the *DataStage Administrator Guide*, SC18-9929-02.

Operators cannot use the DataStage Designer and only see released jobs in the DataStage Director. Neither operators nor developers can create protected projects or add anything to them. This configuration option requires that DataStage run in impersonation mode. This is the default for a root install but needs to be manually configured for a non-root install.

On UNIX installations of DataStage, you can, if required, use an authentication mechanism other than the standard UNIX one. To this end, DataStage supports pluggable authentication modules (PAM). This provides a way of keeping underlying authentication technologies separate from application code, thus eliminating the need to update the application every time that the authentication mechanism is changed. To use PAM authentication, configure DataStage after

you install. For more information, see “Configuring for Use with PAM” in the *DataStage Install and Upgrade Guide*, Part No. 00D-018DS60.

4.21.2 Windows user configuration

On Windows 2003 Server, DataStage must be installed by a user who has local administrator rights. This user must also have read/write permission to the target directory used to install the DataStage server.

If you are logged into a domain account, it must be part of the local administrator’s group on the server that you are installing, and you must have network access to the Windows domain controller for authentication.

The administrative user that you are logged into during installation of DataStage will become the owner, or administrator, of the DataStage installation.

After installation of DataStage, you can use the Windows user management tools to define groups of associated users to restrict access to individual DataStage projects. By default, every login user (the *everyone* group) is given access to newly created projects. However, you can use the Windows user management tools to create multiple Windows groups with assigned users, and then use the DataStage Administrator to restrict access on a per-project basis. For more information, see “User Roles on UNIX Systems” in the *DataStage Administrator Guide*, SC18-9929-02.

4.22 C++ compiler and runtime library requirements

To develop and deploy DataStage parallel jobs, you need to install the C++ compiler that is specific to your platform. Every DataStage development server must have this specific C++ compiler installed (in order to compile BuildOp components or parallel jobs with transformer stages).

For deployment (production) systems, a C++ compiler is not required. However, certain platforms might require that C++ runtime libraries be installed.

Always double-check the specific C++ compiler versions and any associated patches that are required for your specific operating system platform. This information is contained in the DataStage release notes.

Many compilers are licensed on a per-user basis. DataStage only invokes the C++ compiler when the developer compiles a parallel job with transformer stages or compiles a BuildOp component. That is, the maximum number of

simultaneous DataStage developers determines the maximum number of concurrent C++ compiler licenses.

4.22.1 Development systems

To develop parallel jobs, you need the C++ compiler specific to your platform. When installing the C++ compiler for your machine, ensure that all packages are installed.

Important: Only the following compilers and versions are compatible with DataStage. IBM certifies DataStage for specific compiler releases for a given platform.

Table 4-27 lists the supported compilers.

Table 4-27 Supported C++ compilers

Platform	Required C++ compilers
AIX	Visual Age 6, XL C++ Enterprise Edition v7.0 or v8.0
HP-UX (Itanium)	HP Itanium ANSI C++ 6.0
HP-UX (PA-RISC)	HP ANSI C++ A.03.63
LINUX - Red Hat AS 4 32 bit	GCC 3.4
LINUX - Red Hat AS 4 64 bit	GCC 3.4.2
LINIX - Red Hat AS 5 32 bit and 64 bit	GCC 4.1.2
LINUX - SUSE 9.0 kernel 2.6	GCC C++ 3.3.3
Solaris	Sun Studio 10, 11, 12
Windows 2003 Server	Microsoft Visual Studio.NET 2003 (The standalone Microsoft C++ compiler does not work. You must install the complete Visual Studio.NET 2003.)

AIX compiler configuration

An IBM VisualAge® 6 include file might cause some DataStage Paralle Transformer jobs to fail to compile with the following message or similar:

```
"/usr/vacpp/include/stdlib.h", line 123.13: 1540-0040 (S) The text
"undef llabs" is unexpected.  "undef" may be undeclared or ambiguous.
```

This is as per IBM Case 25168:

http://www.developer.ibm.com/tech/faq/individual?oid=2:25168

The solution is to edit the `/usr/vacpp/include/stdlib.h` file, by looking for line 123:

```
undef llabs
```

Replace line 123 with the following text:

```
#undef llabs
```

If you are using the DataStage Parallel pallet and therefore using the IBM VisualAge C++ compiler, you must make sure that the appropriate runtime libraries for the specific version of the compiler are installed on all systems to which parallel jobs will be deployed.

The VisualAge C++ for AIX V6.0 Runtime Libraries can be downloaded from the IBM website at the following address:

http://www.ibm.com/support/docview.wss?rs=0&uid=swg24001467

Red Hat LINUX compiler configuration

Be aware of the following compiler configurations:

- 32-bit RedHat Enterprise Linux Advanced Server 4 requires compiler 3.4.
- 64-bit RedHat Enterprise Linux Advanced Server 4 requires compiler 3.4.2.
- 32-bit and 64-bit RedHat Enterprise Linux Advanced Server 5 require compiler 4.1.2.

To determine the version of the compiler that is currently configured as the default, run the following command:

```
g++ --version
```

4.22.2 Deployment systems

For systems where you are only required to run pre-compiled parallel jobs, a full C++ compiler is not required, but certain runtime libraries are required.

Platforms not mentioned in this section already include the required runtime libraries as part of the operating system, and no further action is required (as long as the runtime libraries have not been explicitly removed).

AIX 5.1 and 5.2 C++ Runtime Libraries

The VisualAge C++ Version 6 Runtime Libraries for AIX 5.x. The runtime library filesets for AIX 5.x can be installed on AIX 5.x using the AIX system administrative tool smit. The libraries are available from the March 2003 PTF at:

http://www.ibm.com/support/docview.wss?rs=0&q=xlC.rte&uid=swg24004427&loc=en_US&cs=utf-8&cc=us&lang=en

Red Hat LINUX Runtime Libraries

On Red Hat LINUX, the legacy runtime libraries can be installed separately from the install CDs. Insert Install CD1 and accept the prompt to autorun. Under the Development category select the **Legacy Software Development** sub-option. Continue with **Forward** to install the libraries.

4.23 Checking product release notes

With any release of DataStage always check the release notes, which are included on the server installation CD in the `readme` directory. Use the server-installation media, because the client installation might have an older release note directory.

4.24 Installing DataStage/Parallel Framework

When prerequisites are complete, installation of DataStage is a straightforward process, as documented in the *DataStage Install and Upgrade Guide*, Part No. 00D-018DS60. As explained in this section, you must consider several options before running the server installation.

4.24.1 Installing multiple DataStage Servers on UNIX

Starting with release 7.5.1, DataStage supports installing multiple DataStage installations on a single UNIX server. Multiple server installations are not supported on Windows 2003 server environments.

For information about installing, upgrading, and managing multiple server instances, see “Installing and configuring multiple server instances” on page 130.

4.24.2 Installing plug-ins

DataStage provides a number of plug-ins that are used to provide connectivity to particular database and data sources, and to perform other special functions. The main install program offers you a selection of plug-ins. In general, unless you are completely confident of your future connectivity needs, it is a good idea to select all plug-ins on the installation. You can determine the available plug-ins by examining the packages directory on the install media.

If you do not select all the required plug-ins at initial install, you can rerun the install in maintenance mode to select additional ones (see "Reinstalling and Maintenance Menu" in the *DataStage Install and Upgrade Guide*, Part No. 00D-018DS60). You can also download plug-ins from the web and install them using the package installer. See "Installing DataStage Packages" in the *DataStage Administrator Guide*, SC18-9929-02, for details.

Important: Only the plug-ins that are installed by the DataStage Installer (install or maintenance mode) are automatically upgraded with future releases. Also certain plug-ins (especially the PACKs) are not automatically installed into new projects, even if they were installed prior to project creation.

4.24.3 UNIX install requirements

DataStage installation requires that the user performing the installation be logged into the system as *root*, unless a non-root install is being performed.

The DataStage installation requires that the administration user (*dsadm* by default) is configured to run with the *sh* or *ksh* shell. Using an alternative shell such as *csh* will cause failures during the installation.

Important: If you are already using GCI, contact IBM Support.

If the installation program detects plug-ins that were installed at a previous release, it upgrades them automatically. If the installation program cannot detect plug-ins that were installed (this is typically because they have been installed by the **`dspackinst`** command rather than by the server installation program), these plug-ins will not be upgraded automatically. These plug-ins can still be upgraded by the installation program, but the user must select them specifically.

Back up existing DataStage projects before attempting to upgrade DataStage. We suggest that you use the DataStage Manager client to export all job designs and definitions from each project. This can be achieved by selecting the **Whole project** radio button in the Export dialog when exporting jobs.

4.24.4 Windows installation requirements

DataStage installation on Windows 2003 Server requires that the user performing the installation be logged into the server as the local administrator, or a domain user with local administrator rights for the server.

When installing DataStage Server on Windows 2003 Server you must have a valid network connection when attached to a domain. If your primary domain controller (PDC) cannot be found across the network, it causes setup to fail when licensing the DataStage engine. If you experience problems, install the product by using a local admin on the machine, or be sure that you are connected to the network. Workgroup users are not affected.

There are known issues when using DataStage on Windows 2003 and also using InfoSphere Information Server 6.0. Symptoms include clients experiencing disconnections and connection error 81022, and occasionally these can occur when InfoSphere Information Server 6.0 is not running. If you are experiencing these symptoms, using either of the following workarounds prevents them from occurring:

- Set the DSRPC service to Interact with the Desktop, specified in the Log On properties of the DSRPC Service in the Services administrative control panel.
- Set Microsoft InfoSphere Information Server 6.0 to run in InfoSphere Information Server 5.0 isolation mode.

The InfoSphere Information Server 5.0 isolation mode is the preferred workaround, but if InfoSphere Information Server 6.0 features are required, then Interact with Desktop mode must be used.

4.25 Verifying the installation log file

On UNIX Systems, always verify the installation log file after a DataStage installation. It might containt error messages that are not captured by the installer. Although it is always a good idea to review the DataStage installation log file, if the UNIX installer appears to hang (that is, if a long list of periods is printed on the console but no progress appears to be made), the installation log file most likely has an error condition.

Text output generated during the installation process is written to a disk-based log file in the `/tmp/dsinstall/logfiles` directory. Each log file contains information specific to a particular installation instance. Log files are generated for both clean and upgrade installations, as well as maintenance mode functions. These log files can be viewed during the installation process using standard UNIX tools, such as tail.

Log file names are of the form `dsinstall.log.MMDDYY.HHMMSS`, where:

- *MMDDYY* equals the two-digit month, day, and year.
- *HHMMSS* equals the two-digit hour, minute, and seconds in a 24-hour format.

Consider the following example:

```
dsinstall.log.082201.162431
```

Using standard UNIX utilities to examine the install log file, perform a case-insensitive search for keywords such as *error* and *warning*.

Important: The DataStage installer creates temporary files and stores settings in the `/tmp/dsinstall` directory. If you need to restart a failed installation, it is best to remove this directory before re-running the DataStage installer.

4.26 Installing DataStage patches

In general, install the official maintenance release (distributed on CD). If you encounter a problem, contact IBM InfoSphere Information Server support to see whether a resolution (for example, a workaround or patch) is available.

In certain instances, when technical issues are discovered after a release of DataStage, patches might be available to correct the problems before the next official maintenance release (which includes fixes for issues discovered after the previous release).

Patches are specific to the version of DataStage for a particular platform (OS/hardware) and are available through IBM InfoSphere Information Server support. A patch is identified by an *eCase* (engineering case) number.

Because there is no automated delivery or update mechanism for patches to DataStage, maintain a list of patches provided to you by IBM support for future installations of a specific release.

Patch list for InfoSphere Information Server 8.1

Fix packs, patches, and refreshed install packages might be available for InfoSphere Information Server and its components. Periodically check the following web page for the current listing:

http://www.ibm.com/software/data/infosphere/support/info-server/download.html

4.27 Installing and configuring optional components

DataStage provides additional connectivity and functionality through new stages, which can be provided through a number of APIs (such as plug In, BuildOp, and Operators), and Packaged Application Connectivity Kits (DataStage *PACKs*).

Each component type provides different ways of installation and configuration. If you have purchased or built any optional components, consult their documentation for installation and configuration.

4.28 Configuring post-installation operating system settings

Certain operating systems (notably Windows 2003 Server) require additional configuration after DataStage has been installed. Where applicable, these post-install configurations are detailed in this section.

Note: Install DataStage first, and then perform the post-installation configuration as outlined in this section.

4.28.1 Securing JobMon ports

Entries should be made in the `/etc/services` file to protect the sockets used by the job monitor. The default socket numbers are 13400 and 13401, and entries in this file might look similar to the following:

- 13400 tcp dsjobmon
- 13401 tcp dsjobmon

These entries prevent use of these sockets by applications other than the job monitor.

4.28.2 Post-installation configuration of Windows 2003 Server

Windows 2003 Server uses different security policies from those used by other Windows platforms. To allow access to a DataStage server on this platform, additional setup steps are required, as outlined below.

You must be logged on as an administrator to be able to set the following access and permissions.

Altering the account for DataStage services

By default a Windows server runs the DataStage services using the LocalSystem account, but you can set up another user to run the services if required. The DataStage services are:

- DSRPC
- DataStage engine resource
- DataStage Telnet

To alter the user associated with these services:

1. Open the Services dialog box in the control panel.
2. In the Properties dialog box for each of the DataStage services, go to the **Log On** tab.
3. Under Log on as, select the **This account** option to specify the user and supply the password.

The user that you choose must have the following privileges:

- Log on local.
- Act as part of the OS.
- Replace a process level token.
- Create a token object.

Two sets of instructions are given below, depending on whether the Windows 2003 Server is a domain controller.

Configuring the Nondomain Controller Windows 2003 Server post-installation

To configure the Nondomain Controller Windows 2003 Server after installation:

1. Allow log on locally:
   a. In Explorer, select **Control Panel** ∅ **Administrative Tools**.
   b. Start Local Security Policy.
   c. Select **Local Policies** ∅ **User Rights Assignment**.
   d. Select **Allow Log on Locally**.
   e. Select **Action** ∅ **Properties** from the menu.
   f. Select **Add User or Group**.
   g. Click **Locations**, the select your local machine.
   h. Click **OK**.
   i. Click **Advanced**.
   j. Click **Find Now**.
   k. Click **Authenticated Users**, then select **OK**.
   l. Click **OK**.
   m. Click **OK**.
   n. Exit the Local Security Policy.

2. Create group:
    a. In Explorer, select **Control Panel** ∅ **Administrative Tools**.
    b. Start **Computer Management**.
    c. Select **System Tools, Local Users & Groups**.
    d. Select **Groups**.
    e. Select **Action** ∅ **New Group** from the menu.
    f. Enter a name for the group (for example, `DataStage Users`).
    g. Click **Create**.
    h. Click **Close**.
3. Add users:
    a. In Explorer, select **Control Panel then Administrative Tools**.
    b. Start **Computer Management**.
    c. Select **System Tools** ∅ **Local Users & Groups**.
    d. Select **Groups** in the tree.
    e. Select the required group (for example, DataStage Users).
    f. Select **Action** ∅ **Add To Group** from the menu.
    g. Select **Add**.
    h. Select **Locations**.
    i. Select your local machine name, then click **OK**.
    j. Select **Advanced**.
    k. Select **Find Now**.
    l. Select users to be added to the group, including authenticated users, then click **OK**.
    m. Click **OK**.
    n. Click **OK**.
    o. Close Computer Management.
4. Set permissions on the DataStage folder:
    a. In Explorer, locate the DataStage solder (for example, `c:\Ascential\DataStage`)
    b. Select **File** ∅ **Properties** from the menu.
    c. Select the **Security** tab, then click **Add**.
    d. Select **Locations**.
    e. Select your local machine name, then click **OK**.
    f. Select **Advanced**.
    g. Select **Find Now**.
    h. Select your group name (for example, DataStage users).
    i. Click **OK**.
    j. Click **OK**.
    k. Select your group.
    l. Check the **Modify** and **Write** check boxes in the Allow column.
    m. Click **OK**.

Configuring the Domain Controller Windows 2003 Server post-installation

To configure the Domain Controller Window 2003 Server after installation:

1. Allow log on locally:

   a. In Explorer, select **Control Panel** ∅ **Administrative Tools**.
   b. Start the Domain Security Policy.
   c. Select **Local Policies** ∅ **User Rights Assignment**.
   d. Select **Allow Log on Locally**.
   e. Select **Action** ∅ **Properties** from the menu.
   f. Select **Add User or Group**.
   g. Select **Browse**.
   h. Select **Advanced**.
   i. Click **Find Now**.
   j. Select **Authenticated Users**, then click **OK**.
   k. Click **OK**.
   l. Click **OK**.
   m. Click **OK**.
   n. Exit the Domain Security Policy.

   Repeat these steps for the Domain Controller Security Policy application.

2. Create a group.

   It is not possible to add the *built-in authenticated users* group to a group that we create in steps 2 and 3, so you might prefer to skip to step 4 and use the *authenticated users* group directly.

   a. In Explorer, select **Control Panel** ∅ **Administrative Tools**.
   b. Start Active Directory and Computers.
   c. Select **Users** in the current domain.
   d. Select **Action** ∅ **New** ∅ **Group** from the menu.
   e. Enter a name for the group (for example, DataStage Users).
   f. Leave the Group scope as Global and Group type as Security.
   g. Click **OK**.

3. Add users:

   a. In Explorer, select **Control Panel, Administrative Tools**.
   b. Start Active Directory and Computers.
   c. Select **Users** in the current domain.
   d. Select the required group (for example, DataStage Users).
   e. Select **Action** ∅ **Properties** from the menu.
   f. Select the **Members** tab.
   g. Click **Add**.
   h. Click **Advanced**, then click **Find Now**.
   i. Select users to add to the group (authenticated users not available).
   j. Click **OK**.

k. Click **OK**.
l. Click **OK**.
m. Close the application.

4. Set permissions on the DataStage folder:

a. In Explorer, locate the DataStage folder (for example, `c:\Ascential\DataStage`).
b. Select **File** ∅ **Properties** from the menu.
c. Select the **Security** tab, then click **Add**.
d. Click **Advanced**.
e. Click **Find Now**.
f. Select a group name, such as DataStage Users or Authenticated Users.
g. Click **OK**.
h. Click **OK**.
i. Select your group.
j. Check the **Modify** and **Write** check boxes in the Allow column.
k. Click **OK**.

Cluster or grid configuration

In addition to single-server (SMP) installations, DataStage also supports clustered deployment of parallel jobs, allowing a single parallel job to run across multiple servers. Clustered deployment is supported for UNIX, LINUX, and Windows releases of DataStage Parallel Engine, but not z/OS Edition.

In a clustered scenario, one server is designated the primary or *conductor* node. This is the node that (typically) DataStage clients connect to, and it is also where the Server Edition engine, repository engine, and job monitor components run.

Grid deployments take the clustered configuration one step further, offering totally dynamic configuration and deployment of clustered DataStage parallel jobs. A grid deployment requires the DataStage Grid Toolkit. Its configuration and setup are described in the accompanying documentation.

Note: In a clustered configuration, only parallel components can run across machines. Because Server Edition components run only on the single conductor node, it is better not to include Server Edition components in scalable parallel job designs.

In a clustered configuration, the DataStage Server Edition engine, repository engine, parallel engine, and job monitor must be installed on at least one server, which typically serves as the conductor node at run time. The remaining servers only need the parallel engine components.

Clustered configuration of DataStage

The clustered configuration of DataStage requires the following:

- All servers in the cluster must be running on the same hardware platform and operating system configuration. You cannot run a single parallel job across a combination of UNIX and Windows servers, nor can you run a single parallel job across different UNIX operating systems, or even different release levels of the same operating system.
- DataStage parallel framework is available on each server in the cluster. It can be installed on each server in the cluster, or installed on a shared file system mount point with the same absolute path on all servers.
- DataStage parallel job components are available across all servers in the cluster. Depending on your situation, one of the following will apply:
  - The DataStage project directory is located on a shared file system mount point with the same absolute path on all servers.
  - The environment variable $APT_COPY_TRANSFORM_OPERATOR is set on the first job run for each job, to copy compiled transform objects:

    Any compiled BuildOp objects and custom components are copied into the LIBPATH (LD_LIBRARY_PATH or SHLIB_PATH) of each node on the cluster.
  - If the BuildOp components were defined and compiled within DataStage client tools, then the project is on a shared mount point across servers.
- DataStage is configured to log in to all servers in the cluster (from each direction) using rsh or ssh without requiring a password.
- Users (and their corresponding groups) that will be running jobs must be created on all servers in the cluster, and must have privileges to run rsh (or ssh) to each remote server in the cluster.
- A parallel configuration file is created that specifies node resources for each server (*fastname*) in the cluster.

4.28.3 UNIX cluster configuration

In this section we discuss and describe the UNIX cluster configuration.

Specifying the remote shell

To find rsh on a processing node, the DataStage Parallel engine searches for the following executables in the order shown:

1. $APT_ORCHHOME/etc/remsh (if it exists)
2. /user/lpp/ssp/rcmd/bin/rsh (AIX only)
3. /usr/ucb/rsh

4. /usr/bin/remsh
5. /bin/remsh
6. /usr/bin/rsh

Where $APT_ORCHHOME is the top-level directory of your parallel engine installation.

If the parallel engine does not find your rsh command, you must specify its location. To do so, copy or rename the supplied file:

```
$APT_ORCHHOME/etc/remsh.example to install_dir/etc/remsh
```

This file contains the following shell script:

```
#!/bin/sh
# Example apt/etc/remsh
exec /usr/bin/rsh "$@"
```

As written, this shell script invokes /usr/bin/rsh. Edit the last line of this script to invoke your specific remote shell command. The script should be executable by all users. Use **chmod** to ensure that it is:

```
chmod 755 script-filename
```

Test this by running rsh on each node:

```
rsh nodename uptime
```

Allowing user execution of RSH without a password

This process is performed differently according to the type of system that you are running. For example, you can either edit `/etc/hosts.equiv` or create a `.rhosts` file for each user. In both cases, add the host name of each parallel processing node to `/etc/hosts.equiv` or `.rhosts`, one host name per line. The host name that is included in this file must correspond to the setting of the node's fastname parameter in the parallel configuration file. For information about the fastname configuration option, see the section "Node Names" in the *DataStage Parallel Job Developer's Guide*, LC18-9891.

If you choose to edit the `/etc/hosts.equiv` file, the file must be owned by root and must grant read/write access to root and no access to any other user (file mode of 600).

If you choose to create an `.rhosts` file for each user, it must meet the following criteria:

- Be located in the home directory of each parallel user.
- Be owned by the user.

- Grant read/write access to the user and no access to any other user (file mode of 600).

To check that users can use rsh without a password, issue the following command on each node:

```
rsh hostname uptime
```

Here *hostname* is the name of a processing node that you use with the parallel engine. If hostname is accessible, this command prints a message displaying the time that it has been up.

Configuring the Parallel Framework to use ssh instead of rsh

This assumes that ssh servers have been installed on all the machines where DataStage is to run. SSH needs to be configured such that you can launch a command from the conductor node to all other nodes without a password, but instead authenticate via public key encryption. Carry out the following process for each user that will run DataStage parallel jobs. Assume that the conductor node is etlnode and the remote node is dbnode.

1. Generate a public/private DSA key pair on the conductor, using:

   ```
   etlnode% ssh-keygen -b 1024 -t dsa -f ~/.ssh/id_dsa
   ```

   The identification keys have been saved in ~/.ssh/id_dsa. When you are asked for a passphrase, leave it empty. Now send the public key to the remote node. The **scp** command is the secure version of the **rcp** command, as follows:

   ```
   etlnode% cd .ssh
   etlnode% scp id_dsa.pub user@dbnode:~/.ssh
   ```

2. Log in to dbnode and add the public key to the list of authorized keys using the following commands:

   ```
   dbnode% cd .ssh
   dbnode% cat id_dsa.pub >> authorized_keys2
   dbnode% chmod 640 authorized_keys2
   dbnode% rm -f id_dsa.pub
   ```

   The filename is *authorized_keys2*, not authorized_keys. You should now be able to ssh from etlnode to dbnode without a password. For example, the following should work from etlnode:

   ```
   etlnode% ssh dbnode ls
   ```

3. Create a /apps/Ascential/DataStage/PXEngine/etc/remsh file, which contains:

   ```
   #!/bin/sh
   exec /usr/bin/ssh "$@"
   ```

You can find a similar example of this file at:

```
$APT_ORCHHOME/etc/remsh.example
```

Installing the parallel engine on a remote node on UNIX

If you plan to use DataStage in a cluster across multiple servers, you need to ensure that the parallel engine components are installed on all the nodes. This can be done either of the following ways:

- Install DataStage to a shared file system that is mounted (with the same absolute path) across all servers in the cluster.
- Use the maintenance menu of the DataStage installer to copy over the `$APT_ORCHHOME` directories to new nodes. To use this facility, you must configure rsh (or ssh) access to remote servers before the installation. This feature uses the command-line utility **`orchcopydist`**, which can also be run directly from the UNIX command line.

Configuring AIX clusters

If you are installing the parallel engine on an AIX cluster, you must verify the setting of the network parameter *thewall*. The value of this parameter can greatly affect the performance of the parallel engine.

Set *thewall* to at least 25% of each node's physical memory, or the maximum allowed on your system, if that is less than 25% of memory. The maximum value of *thewall* is AIX-version dependent. The main page for the network option's (**`no`**) command contains the details and system default values. The value of *thewall* is specified in kilobytes. For example, if each node on your system has 256 MB (262,144 KB) of physical memory, set *thewall* to 65,536.

To set *thewall*, use the following steps:

1. Determine the amount of physical memory on a node (the value of *realmem* is the amount of physical memory on the node in KB), by using the following:

   ```
   lsattr -E -l sys0 | grep realmem
   ```

2. Determine the current setting of *thewall* for a particular node or workstation:

   ```
   /usr/sbin/no -a | grep thewall
   ```

3. Set *thewall* by doing one of the following:

   a. On a specific node, execute the following command with root privileges:

      ```
      /usr/sbin/no -o thewall=65536
      ```

   b. Set *thewall* on all nodes of an AIX system by executing the following **`dsh`** command from the control workstation, with root privileges:

      ```
      dsh -a no -o thewall=65536
      ```

4.28.4 Windows cluster configuration

This section highlights the Windows cluster configuration.

Configuring RSH access on Windows 2003 Server

In a clustered configuration, configure rsh on each of the Windows 2003 servers:

1. On each remote player node, you must create a user with the same login ID and password as the user running the job on the conductor node.
2. From the control panel, double-click the System icon, select the **Advanced** tab, and click **Environment Variables**.
3. In the User Variables section, change the value of the HOME environment variable from %HOMEDRIVE%%HOMEPATH% to C:/Documents and Settings/username. Click **OK** to accept the settings, click **OK** to exit from the Environment Variables dialog, and click **OK** to exit from system properties.
4. Open the Notepad application and create a text file called `.rhosts` in the `C:/Documents and settings/username` directory. Edit the file and place the following line of text in the file:

   ```
   + + (press enter <cr> after the + +)
   ```

   There should be a space between the plus signs. Make note of the exact machine names for each of the player nodes, as these names will be needed when defining the multi-node DataStage configuration file.

Enabling rsh authentication on Windows 2003

On the master (conductor) Windows 2003 server, you must enable rsh authentication. The rsh mechanism on the conductor node (usually the same computer running the DataStage server) must be set up for proper authentication of rsh commands as follows:

1. Be sure that you are logged in to Windows with the user ID of the user who will run DataStage jobs.

   Note: Windows domain logins are not supported for clustered deployment of DataStage jobs.

   Local user accounts must be created on each server in the Windows cluster.

2. Run the **`rsetup`** command via the MKS shell.
3. Enter your password.

4. To test that rsh is working to node PC_n, enter the command:

```
rsh PC-n ls
```

This command sends the **ls** command to use the domain and user name of the currently logged-on user.

Installing DataStage parallel engine on a remote node

To configure a clustered deployment of DataStage for Windows, you must install DataStage 8.1 Server for Windows on each of the Windows 2003 servers. Unlike UNIX, there is no maintenance facility (or corresponding **orchcopydist** command) to distribute the installation to remote servers in the cluster.

4.29 Configuring the DataStage environment and default settings

This section highlights the DataStage environment and default settings.

4.29.1 Setting the DataStage environment

DataStage provides a number of operating system environment variables to enable and disable product features and to fine-tune job performance.

Although operating system environment variables can be set in multiple places, there is a defined order of precedence that is evaluated when a job's actual environment is established at run time:

1. The daemon for managing client connections to the DataStage server engine is called *dsrpcd*. By default (in a root installation), dsrpcd is started when the server is installed and should start whenever the machine is restarted (unless this machine is part of a high-availability (HA) configuration). dsrpcd can also be manually started and stopped using the **$DSHOME/uv -admin** command. (For more information, see the *DataStage Administrators Guide*, LC18-9895.)

   By default, non-UNIX System Services DataStage jobs inherit the dsrpcd environment, which on UNIX platforms is set in the `etc/profile`, `$DSHOME/dsenv`, and `ds.rc` scripts. On Windows, the default DataStage environment is defined in the registry. Because client connections are forked from DSRPC, they do not pick up per-user environment settings from their `$HOME/.profile` script.

2. Environment variable settings for particular projects can be set in the DataStage Administrator client. Any project-level settings for a specific environment variable will override any settings inherited from dsrpcd.

Important: When migrating projects between machines or environments, project-level environment variable settings are not exported when a project is exported. These settings are stored in the project directory in the `DSPARAMS` file. Any project-level environment variables must be set for new projects.

Within DataStage Designer, environment variables can be defined for a particular job using the Job Properties dialog box. Any job-level settings for a specific environment variable will override any settings inherited from dsrpcd or from project-level defaults.

4.29.2 Altering the DataStage dsenv on UNIX

On UNIX Systems, the DataStage server has a centralized file for storing environment variables called *dsenv*. It resides in $DSHOME, where $DSHOME identifies the DataStage main directory (for example, `/opt/IBM/InformationServer/Server/DSEngine`).

The `dsenv` file is a series of Bourne shell arguments that are referenced during DataStage server startup and can be referenced by interactive users or other programs or scripts. You are likely to want to add new environment variables as you configure DataStage to connect to different databases using plug-ins or ODBC drivers (see "Configuring plug-ins" and "Configuring ODBC Access" in the *DataStage Install and Upgrade Guide,* Part No. 00D-018DS60).

To emulate the DataStage server environment, in a Bourne shell execute the following command from the `$DSHOME/DSEngine` directory:

```
. ./dsenv
```

You must include any environment variable required by the DataStage server for all projects in the `dsenv` file.

Certain plug-ins require shared libraries to be loaded, and you need to include the library path in an environment variable. The names of the library path environment variables are platform dependent (Table 4-28).

Table 4-28 Library path environment variables

Platform	Library path environment variable
AIX	LIBPATH
HP-UX (PA-RISC)	SHLIB_PATH
HP-UX (Itanium), LINUX, Solaris	LD_LIBRARY_PATH

Note: HP-UX 11.23 shared libraries are suffixed .so instead of .sl.

For any changes to, or addition of, any environment variables in dsenv to become effective, the DataStage server should be stopped and restarted as follows:

1. To stop the server, use the following:

   ```
   $DSHOME/bin/uv -admin -stop
   ```

2. To start the server, use the following:

   ```
   $DSHOME/bin/uv -admin -start
   ```

4.29.3 Suggested default settings for all projects

The following project-level default settings are suggested for all DataStage projects. Default settings can be made through the DataStage Administrator, using the Environment button on the General tab.

```
$APT_DUMP_SCORE (found in the Parallel/Reporting category)
```

By default, the DataStage Job Monitor uses a time-based job monitor that can introduce intermittent problems on certain platforms. Time-based job monitoring can be disabled in favor of size-based job monitoring by altering the following environment variables (in the Parallel category):

```
$APT_MONITOR_TIME = unset
$APT_MONITOR_SIZE = 100000
```

Default settings on AIX

On AIX, the environment variable $LDR_CNTRL is sometimes used (in combination with other variables) to increase available memory to Server Edition jobs (see “Increasing DataStage Server Edition memory on AIX” on page 143). However, setting this variable can cause jobs to fail on AIX.

If you are upgrading an install of DataStage Server Edition or are running both Server and Parallel jobs on AIX platforms, make sure that the environment variable LDR_CNTRL is not set by default in either the *dsenv* file or as a default environment variable in administrator.

Default settings on Solaris

On Solaris platforms, if you need to create and read very large parallel data sets (where the underlying files are greater than 2 GB), you must define the environment variable $APT_IO_NOMAP.

4.30 Configuring the DataStage administrator environment

This section explains how to configure the DataStage administrator environment.

4.30.1 Setting the UNIX and LINUX administrator environments

On UNIX and LINUX systems, a number of command-line utilities are provided to administer the DataStage engine and to manage objects (such as configuration files and parallel data sets) used and created by DataStage.

To use these utilities, the default login profile should be altered for the users who are administering DataStage to include the environment variables listed in Table 4-29.

Table 4-29 Default login profile environment variables

Environment variable	Setting	Example
DSHOME	Identifies top level of the Server Edition directory	`/opt/IBM/InformationServer/Server/DSEngine/`
APT_ORCHHOME	Identifies top level of the parallel engine directory	`/opt/IBM/InformationServer/Server/PXEngine/`
APT_CONFIG_FILE	Location of the specified parallel configuration file	`/opt/IBM/InformationServer/Server/Configurations/default.apt`
PATH	Search path for executable directories	`$PATH:$DSHOME/bin:$APT_ORCHHOME/bin`
LIBPATH or LD_LIBRARY_PATH or SHLIB_PATH	Search path for object libraries	`$LIBPATH:$DSHOME/lib:$APT_ORCHHOME/lib`

4.30.2 Setting the Windows 2003 environment

On Windows 2003, environment variables are set on a per-user or system-wide basis through the Windows control panel. To alter or create new environment variables on a Windows 2003 server:

1. From the Windows Start menu, select **Settings** ∅ **Control Panel**.
2. From the control panel, select **System** to display the system dialog.
3. From the System dialog, select the **Advanced** tab.
4. Select **Environment Variables** to display the Environment Variables dialog.
5. To edit an existing environment variable, select the variable and click **Edit**.

6. To add a new environment variable, select **New**.
7. To delete a new environment variable, select **Delete**.
8. Click **OK** to accept the changes.

4.31 Configuring and verifying database connectivity

DataStage Parallel palette provides a number of pre-built components (stages) for accessing data in popular Relational Database Management Systems (RDBMS).

On most platforms, native parallel connectivity is provided to IBM DB2 (with DPF), IBM Informix®, ODBC, Oracle, Sybase, and Teradata. Specific configuration instructions are provided in the *DataStage Install and Upgrade Guide*, Part No. 00D-018DS60.

In addition, some database connectivity is provided through DataStage plug-ins. Documentation for installing and configuring plug-ins is provided in the *DataStage Plug-in Installation and Configuration Guide*. See "Installing DataStage Packages" in the *DataStage Administrator Guide*, SC18-9929-02, for details.

In cases where DataStage is deployed as a 32-bit application, 32-bit database libraries are used. Otherwise, 64-bit libraries are used.

Where necessary, this section provides supplemental database configuration instructions that are not in the install guides.

4.31.1 DB2 configuration for Enterprise stage

To connect to DB2 (DB2 Enterprise Server Edition with DPF) on UNIX, you must install or crossmount DataStage Parallel engine on each node of the DB2 cluster. In essence, you are creating a DataStage cluster, as described in “Cluster or grid configuration” on page 93.

Note: Specific, detailed instructions for configuring DataStage against a DB2 database are included in “Configuring remote DB2” on page 131.

To connect to DB2 databases using DataStage, run the script **`$APT_ORCHHOME/bin/db2setup.sh`** from the UNIX command line to configure DataStage to access the DB2 database. The file must be called once for each DB2 database to be accessed by DataStage users. Pass the database name as

an argument. For example, the following command calls db2setup.sh to configure DataStage to access the database db2_8:

```
db2setup.sh db2_8
```

You must grant privileges to each user who will run jobs containing the DB2 Enterprise Stages by running the following script once for each user:

```
$APT_ORCHHOME/bin/db2grant.sh
```

You need DBADM privileges to run this script. The syntax of the command is:

```
db2grant.sh database_name user_name
```

DB2 configuration for z/OS

DB2 for z/OS Versions 7.1 and 8.1 are supported by DataStage. The following are considerations and actions to be taken, depending on your particular z/OS environment:

- The DSNTIJCL job from the SDSNSAMP library must be run. This job binds the default CLI packages and plan needed by the DB2 stage.
- For DB2 for z/OS Version 8, PTF UQ89056 (from APAR PQ88085) must be applied. This PTF defines a character set conversion between 367 and 1208, which is needed for Version 8.
- Make sure that these exits have been implemented (DSN3@SGN, DSN3@ATH, and DSNX@XAC) if access to DB2 is controlled via IBM RACF®. Refer to the *DB2 Administration Guide* for your version of z/OS.
- Authorization to use the DB2 load utility if that is to be used with DataStage. Generally, if you are running this from the IBM MVS™ environment today, you should be fine.
- The DB2 bind needs to be run to allow DataStage connection. For this, refer to the DataStage installation instructions.
- If DB2 loads are to be done, verify the applicability of DB2 resource definition changes in the configuration file (to set the high-level qualifier (HLQ) for the temporary datasets).

4.31.2 Informix configuration

Testing has shown that it is not possible to configure a UNIX environment to connect to both Sybase ASE and Informix IDS databases at the same time. This means that you cannot construct a job that contains both the Sybase OC stage and the Informix CLI stage.

In addition, specific configuration of the environment needs to be done for the Informix CLI stage to connect to an Informix IDS database. You must ensure that the following files are configured correctly:

- etc/hosts
- etc/services
- $INFORMIXDIR/etc/sqlhosts
- $DSHOME/.odbc.ini
- $DSHOME/../Projects/<Project Name>/uvodbc.config

Also, in the $DSHOME/dsenv file, verify that the environment variables are set appropriately:

- INFORMIXDIR
- INFORMIXSERVER
- INFORMIXBIN
- INFORMIXC
- THREADLIB

The PATH environment variable should have $INFORMIXDIR/bin appended to it. The environment variable LIBPATH (LD_LIBRARY_PATH or SHLIB_PATH on some platforms) should have $INFORMIXDIR/lib, $INFORMIXDIR/lib/cli, and $INFORMIXDIR/lib/esql appended to it. Ensure that these three are after the DataStage directories in LIBPATH (LD_LIBRARY_PATH or SHLIB_PATH).

Informix configuration on AIX

The following is an example of the settings in the $DSHOME/dsenv file for AIX:

```
#Informix sdk 2.8 UC2-1
#
INFORMIXSERVER=<server name>; export INFORMIXSERVER
#
INFORMIXDIR=<Informix directory path>; export INFORMIXDIR
LIBPATH=`cat
/.dshome`/lib:$LIBPATH:$INFORMIXDIR/lib:$INFORMIXDIR/lib/cli:
$INFORMIXDIR/lib/esql; export LIBPATH
INFORMIXC=CC; export INFORMIXC
THREADLIB=POSIX;export THREADLIB
PATH=$PATH:$INFORMIXDIR/bin; export PATH
```

Note: The /.dshome file might not exist in iTag installations (see “Installing and configuring multiple server instances” on page 130). In these instances, you might need to explicitly set $DSHOME.

Within your INFORMIXSERVER definition, set the protocol to onsoctcp.

Informix configuration on Solaris

On Solaris, when DataStage is installed, additional configuration needs to be performed to avoid conflicts. Following is an example `$DSHOME/dsenv` file. Note the sequence of entries in the LD_LIBRARY_PATH:

```
1. #Informix sdk 2.8 UC1
2. #
3. INFORMIXSERVER=<server name>; export INFORMIXSERVER
4. #
5. INFORMIXDIR=<Informix directory path>; export INFORMIXDIR
6. INFORMIXBIN=$INFORMIXDIR/bin; export INFORMIXBIN
7. LD_LIBRARY_PATH=$LD_LIBRARY_PATH:$INFORMIXDIR/lib:$INFORMIXDIR/lib/cl
   i:
$INFORMIXDIR/lib/esql:$APT_ORCHHOME/lib:$APT_ORCHHOME/DSCAPIOp
:
$APT_ORCHHOME/osh_wrappers:$APT_ORCHHOME/usr_osh_wrappers:
$APT_ORCHHOME/etc; export LD_LIBRARY_PATH
8. INFORMIXC=CC; export INFORMIXC
9. THREADLIB=POSIX;export THREADLIB
10.PATH=$PATH:$INFORMIXDIR/bin; export PATH
```

Within your INFORMIXSERVER definition, the protocol should be set to *ontlitcp*.

4.31.3 Oracle configuration for Enterprise stage or connector

To configure the Oracle Enterprise stage with DataStage, install and configure the Oracle Database Utilities and Oracle Network software on your DataStage server.

You must also complete the following step (often these are set in the `dsenv` file):

1. Create the user-defined environment variable ORACLE_HOME and set this to the $ORACLE_HOME path (for example, /disk3/oracle9i).
2. Create the user-defined environment variable ORACLE_SID and set this to the correct service name (for example, ODBCSOL).
3. Add ORACLE_HOME/bin to your PATH and ORACLE_HOME/lib to your LIBPATH, LD_LIBRARY_PATH, or SHLIB_PATH.

In addition, for Oracle connectivity you must:

- Have login privileges to Oracle using a valid Oracle user name and corresponding password. These must be recognized by Oracle before you attempt to access it.
- Have SELECT privilege on:
  - - DBA_EXTENTS
  - - DBA_DATA_FILES
  - - DBA_TAB_PARTITONS
  - - DBA_TAB_SUBPARTITIONS
  - - DBA_OBJECTS
  - - ALL_PART_INDEXES
  - - ALL_PART_TABLES
  - - ALL_INDEXES
  - - SYS.GV_$INSTANCE

    Only if Oracle Parallel Server is used. If not, you might need to specify the environment variable APT_ORACLE_NO_OPS to prevent OPS checks from being performed.

Note: `$APT_ORCHHOME/bin` must appear before `$ORACLE_HOME/bin` in your `$PATH`.

To ease administration, you can create a role that has the appropriate SELECT privileges, as follows:

```
CREATE ROLE DSXE;
GRANT SELECT on sys.dba_extents to DSXE;
GRANT SELECT on sys.dba_data_files to DSXE;
GRANT SELECT on sys.dba_tab_partitions to DSXE;
GRANT SELECT on sys.dba_tab_subpartitions to DSXE;
GRANT SELECT on sys.dba_objects to DSXE;
GRANT SELECT on sys.all_part_indexes to DSXE;
GRANT SELECT on sys.all_part_tables to DSXE;
GRANT SELECT on sys.all_indexes to DSXE;
```

After the role is created, grant it to users who will run parallel DataStage jobs, as follows:

```
GRANT DSXE to <oracle userid>;
```

Oracle configuration on AIX

On the AIX platform, users must have SELECT access to the sys.gv_$instance and sys.v_$cache tables. Issue the following SQL statements to grant this access:

```
GRANT select ON sys.gv_$instance TO public;
GRANT select ON sys.v_$cache TO public;
```

If you do not have Oracle OPS on these platforms, set the $APT_ORACLE_NO_OPS environment variable to disable OPS checking on the Oracle Enterprise stage.

Oracle configuration on HP-UX

By default, the DataStage `dsenv` file will be installed with support for Oracle 9i and 10g connectivity. If support for an Oracle 8i client is required, the environment must be modified to something similar to the following. PA_RISC2.0 must be replaced by PA_RISC:

```
SHLIB_PATH=`dirname
$DSHOME`/branded_odbc/lib:$DSHOME/lib:$DSHOME/uvdlls

  $DSHOME/java/jre/lib/PA_RISC:$DSHOME/java/jre/lib/PA_RISC/hotspot

  $ORACLE_HOME/lib;export SHLIB_PATH

SHLIB_PATH=:$DSHOME/java/jre/lib/PA_RISC:

  $DSHOME/java/jre/lib/PA_RISC/hotspot:$SHLIB_PATH; export SHLIB_PATH
```

To run Oracle9i or 10G, you need:

```
SHLIB_PATH=`dirname
$DSHOME`/branded_odbc/lib:$DSHOME/lib:$DSHOME/uvdlls

  $DSHOME/java/jre/lib/PA_RISC2.0:$DSHOME/java/jre/lib/PA_RISC2.0/hotsp
ot Export SHLIB_PATH
```

To run Oracle 9i on HP-UX 11i, you must add $ORACLE_HOME/lib32 preceding $ORACLE_HOME/lib in the SHLIB_PATH environment variable.

4.31.4 Sybase configuration

The Sybase operators are built against Version 12.5 of the Sybase Open Client. If you are using an earlier version of the Sybase Open Client, you will need to upgrade. Version 12.5 of Sybase Open Client is compatible with earlier versions of the Sybase ASE server.

Using Version 12.5 of the Sybase Open Client, DataStage 7.5.2 also supports release 15 of Sybase ASE with ASE 12.5 level of functionality (new ASE 15 features are not supported). If you are using the Sybase ASE 15 client, a configuration change is required, as documented in this section.

Configuring Sybase Open Client 12.5

Sybase open client software has to be installed on the DataStage server side. The configuration details are:

1. Create the user-defined environment variable SYBASE and set this to the $SYBASE path that specifies the Sybase home directory (for example, export SYBASE=/disk3/Sybase).
2. Create the user-defined environment variable SYBASE_OCS and set this to the Sybase open client software installation directory (for example, export SYBASE_OCS=OCS-12_5).
3. Interfaces file: Add the details about the database server (database name, host machine name or IP address, and port number) to the interfaces file located in the `$SYBASE` directory.
4. Add SYBASE/bin to your PATH and SYBASE/lib to your LIBPATH, LD_LIBRARY_PATH, or SHLIB_PATH.
5. Have login privileges to Sybase using a valid Sybase user name and corresponding password, server name, and database. These must be recognized by Sybase before you attempt to access it.

Note: $SYBASE/$SYBASE_OCS/bin must appear first in your PATH. This is to ensure that the script $SYBASE/$SYBASE_OCS/bin/isql is always executed when the user runs the **`isql`** command.

When accessing Sybase Databases with NLS, the following steps are required:

1. Create a database using a collation of the language that you are going to test (for example, create database <<database path>> COLLATION 932JPN for Japanese (Shift_JIS) database).
2. Install the DataStage server in that particular language (for example, Japanese (Shift_JIS)). Upgrading the existing DataStage server will not work, as you will not have an option to select support of other languages. You need to uninstall the existing server and install it with the language that you want.
3. The language that you want to test should be the default setting on your OS (desktop) (that is, the machine on which you are going to test through the DataStage client). Select the language through **Control Panel** ∅ **Regional settings**, and the keyboard input should also be set to that language.

Client setting: Using the NLS tab in the Enterprise stage, you must select the language that you want to test. For example, if your OS has Japanese as the default, then in the DataStage client the project default will be Shift_JIS, which you do not need to select for every job that you run.

Configuring the Sybase ASE 15 client on UNIX

Changes in the naming of Sybase ASE 15 client libraries make this release incompatible with DataStage when used directly. It is suggested that you install Sybase Open Client 12.5, which is compatible with Sybase ASE 15 and earlier.

With the following workaround on the DataStage server machine, it is possible to configure Sybase ASE 15 client to work with DataStage 7.5.x.

Create symbolic links for the Sybase 15 library files in the /OCS-15_0/lib/*.so path (so that they appear to DataStage 7.5.2 as having the previous expected names):

- libct.so ∅ libsybct.so
- libtcl.so ∅ libsybtcl.so
- libcs.so ∅ libsybcs.so
- libcomn.so ∅ libsybcomn.so

Configuring the Sybase ASE 15 client on Windows

Changes in the naming of Sybase ASE 15 client libraries make this release incompatible with DataStage when used directly. It is suggested that you install Sybase Open Client 12.5, which is compatible with Sybase ASE 15 and earlier.

With the following workaround on the DataStage server machine, it is possible to configure Sybase ASE 15 client to work with DataStage 7.5.x.

Rename the Sybase 15 ASE Open Client Libraries as follows (to reflect their previous names at release 12.5):

- libct.dll ∅ libsybct.dll
- libtcl.dll ∅ libsybtcl.dll
- libcs.dll ∅ libsybcs.dll
- libcomn.dll ∅ libsybcomn.dll

4.31.5 Teradata configuration for Enterprise Stage

You must install the Teradata Utilities Foundation on all nodes that will run DataStage parallel jobs. Refer to the installation instructions supplied by Teradata. (You need system administrator status for the install.)

You must set up a Teradata database user (this is the user who will be referred to by the DB options property in the Teradata stage). The user must be able to create tables and insert and delete data. The database for which you create this account requires at least 100 MB of PERM space and 10 MB of SPOOL. Larger allocations might be required if you run large and complex jobs. You need database administrator status to create the user and database.

The example that follows shows you how to create the orchserver account. The user information is stored in the terasync table. The name of the database in this example is userspace. The following four commands for BTEQ are used to set up the account:

- `CREATE USER orchserver FROM userspace AS`
- `PASSWORD = orchserver`
- `PERM = 100000000`
- `SPOOL = 10000000`

Once the account is set up, issue the following command:

```
GRANT select ON dbc TO orchserver;
```

Creating a database server

If you want to use a pre-existing Teradata user, you only need install a database server and configure it to use a new database. Install the new database server with the same PERM and SPOOL values as shown above.

The following is an example of creating a database server called devserver using table userspace:

```
CREATE DATASBASE devserver FROM userspace AS
PERM = 100000000
SPOOL = 10000000
GRANT create table, insert, delete, select ON devserver TO orchclient;
GRANT create table, insert, delete, select ON devserver TO orchserver;
```

Teradata is optimized for batch operations with large volumes of data. Teradata offers multiple utilities for extracting and loading data, with differing usage rules. For this reason, DataStage offers multiple Teradata stages to choose for the source and target of a given data flow. All data flows must be constructed on the parallel canvas within the DataStage. It is important to understand that although you are deploying parallel jobs, all stages from a Teradata perspective will not run in parallel.

The Teradata Enterprise stage is intended for maximum parallel performance for sources or targets in parallel. This interface is flexible, and for Teradata instances with a large number of AMPs (VPROCs), it might be necessary to set the

optional SessionsPerPlayer and RequestedSessions in the DBOptions string in the Teradata Enterprise stage.

4.31.6 Netezza connectivity

DataStage supports Netezza® Performance Server (NPS®) targets on AIX, Linux Red Hat, Linux SuSE, and Solaris platforms.

Documentation for the Netezza Enterprise stage is installed with the DataStage client referenced in the documentation bookshelf.

The Netezza Enterprise stage is a write stage. The stage takes bulk data from a data source and writes that data to a specified destination table in NPS. You can write data to NPS using two available load methods (Table 4-30).

Table 4-30 Netezza load methods

Load method	Description	Requirements
Netezza load	Uses NPS nzload utility to load directly to target NPS table.	LOAD privileges for the target table. Data in the source database is consistent, contains no default values, has single-byte characters only, and uses a predefined format.
External table	Writes to an external table within NPS. Data is then streamed into the target table.	If the data source contains default values for table columns and uses variable format for data encoding such as UTF-8.

To write data to NPS using the Netezza Enterprise stage, you must install the required components. Then you must configure the stage and your system appropriately. The following is a list of the installation and configuration requirements:

- Install the Netezza server and client. The Netezza Enterprise stage supports Netezza Server 2.05 Patch 4 and later. You must install the Netezza client if you want to use the nzload load method.
- Install and configure the nzload utility and ODBC driver provided by NPS. The Netezza Enterprise stage uses this ODBC driver to retrieve metadata for the destination table or file. Ensure that you install the 2.05 version of this ODBC driver. This driver is in conformance with 3.0 ODBC specifications.
- Install and configure DataDirect's ODBC driver manager (see 4.32, “Configuring and verifying ODBC connectivity” on page 114).
- Obtain explicit LOAD privileges for the target table in NPS. For more information or help for installing and configuring the above, see the documentation that accompanies the above software.

- Create the odbc.ini file in $DSHOME. To create the odbc.ini file, you must first configure the .odbc.ini file located in $DSHOME by adding necessary information, such as database name, host name, user name, and password. Then copy the contents of the .odbc.ini file to the odbc.ini file in the same location. Alternatively, you can create a soft link to the .odbc.ini file.

First, add the following entries in the .odbc.ini file:

```
1. ODBC Data Sources]
2. NZSQL = NetezzaSQL
3. NZSQL]
4. Driver = [Enter the driver path]
5. Description = NetezzaSQL ODBC
6. Servername =
7. Port = 5480
8. Database = [Enter the database name.]
9. Username = [Enter the user name to connect to the database.]
10.Password = [Enter the password to connect to the database.]
11.CancelAsFreeStmt = false
12.CommLog = false
13.Ksqo = false
14.Lie = false
15.Optimizer = false
16.Parse = false
17.PreFetch = 256
18.Protocol = 7.0
19.ReadOnly = false
20.ShowSystemTables = false
21.Socket = 16384
22.DateFormat = 1
23.TranslationDLL = [Enter the appropriate variable value.]
24.TranslationName = [Enter the appropriate variable value].
```

Now add the environment variable NZ_ODBC_INI_PATH and have it point to the parent directory of the odbc.ini file. This odbc.ini file can be a copy of, or a soft link to, the .odbc.ini file.

Set user-defined and general environment variables appropriately. Table 4-31 on page 114 provides information about how to set user-defined and general environment variables for Netezza.

Table 4-31 Netezza environment variables

Environment variable	Setting	Description
$NETEZZA	[path]	Specifies the Netezza home directory.
$NZ_ODBC_INI_PATH	[filepath]	Points to the location of the `.odbc.ini` file.
$PATH	Should include $NETEZZA/bin	Search the path for executable files.
$LIBPATH, $SHLIB_PATH, or $LD_LIBRARY_PATH	Should include $NETEZZA/lib	The actual environment variable name depends on the platform to set the search path for library files.

4.32 Configuring and verifying ODBC connectivity

In this section we describe how to configure and verify ODBC connectivity for Enterprise stage and connector.

DataStage provides access to any database that supports ODBC using database-specific ODBC drivers, which are included on the installation media.

The ODBC Drivers are an OEM version of the Data Direct ODBC Driver package. These drivers are licensed solely for use with DataStage and require certain connection parameters to be set in order to function properly. Do not try to use these drivers with other applications, as licensing errors will occur.

ODBC driver packs are often updated between major releases of DataStage. You are strongly advised to check your release notes for more up-to-date information about ODBC drivers.

The ODBC drivers are one of two types, depending on the database being connected to and your platform type:

- Non-wire protocol drivers. These drivers require you to install the database-specific client software for the database on the DataStage server. (The drivers use the API supplied by the database client.)
- Wire protocol drivers. These drivers do not require database client software. (They communicate with the database directly.)

Important: A newer version (v5.1/6.0) of the bundled DataDirect ODBC Drivers might be available for download through the IBM eService support site (as part of your support contract). Check this site to verify availability for your platform.

4.32.1 Configuring ODBC access on UNIX

On UNIX systems, the DataDirect ODBC drivers are installed in the `$DSHOME/../branded_odbc` directory. These drivers must be installed and configured before they can be used by DataStage.

You will need to edit three files to set up the required ODBC connections, and they are as follows:

- `dsenv`
- `.odbc.ini`
- `uvodbc.config`

All three files are in the `$DSHOME` directory.

In addition, each DataStage project has a `uvodbc.config` file that can be used to override or extend the `uvodbc.config` settings in `$DSHOME`. This allows access to be customized by project for security and virtualization requirements.

Non-wire drivers require different setup information to wire drivers. Non-wire drivers require information about the location of the database client software, and wire drivers require information about the database itself.

Within your `dsenv` file:

- The shared library path should be modified to include `$DSHOME/../branded_odbc/lib`.
- The ODBCINI environment variable will be set to `$DSHOME/.odbc.ini`.
- Add $APT_ORCHHOME/branded_odbc to your PATH.
- Add $APT_ORCHHOME/branded_odbc /lib to your LIBPATH, LD_LIBRARY_PATH, or SHLIB_PATH.
- The ODBCINI environment variable must be set to the full path of the `odbc.ini` file (which by default is the hidden `$DSHOME/.odbc.ini` file, but can be any syntactically correct file).

Further details on ODBC configuration can be found in the *DataStage Install and Upgrade Guide*, Part No. 00D-018DS60.

4.32.2 Setting up DSNs on UNIX

The DataDirect Drivers Reference manuals provide specific information for configuring the ODBC environment for a particular data source. These manuals are installed on your DataStage server in the `$DSHOME/../branded_odbc/books` path.

Copy these files (which are in PDF format) from the server to your client machine for review.

4.32.3 Configuring ODBC access on Windows 2003 Server

The ODBC driver pack that ships with the Windows version of DataStage is available as a separately installable module. The drivers are located in the `DataStage ODBCDrivers` directory on the DataStage CD.

4.32.4 ODBC readme notes

After installation refer to the `readmeODBC.txt` file located in the `branded_odbc` directory (UNIX) or the `drivers` directory (Windows) for further information regarding ODBC driver configuration and use. This file is also linked to the HTML release notes.

4.33 Creating and verifying project location

By default, DataStage Administrator creates its projects (repositories) within the projects directory of the DataStage installation directory.

In general, do not create DataStage projects in the default directory for the following reasons:

- Disk space is typically limited in product installation file systems.
- Backup and restore policies on product installation file systems are often less frequent than user or database directories.
- Standards for many production environments dictate that product installation directories should not be used for actual development storage.
- With a default project location, careless developers can mistakenly fill up the file system containing DataStage and corrupt not only their project, but also other projects and components in the DataStage installation directory.

For these reasons, specify a different file system (directory) when creating new DataStage projects.

DataStage projects should be stored outside of the installation directory on a redundant file system with sufficient space (minimum 100 MB per project). For cluster/grid implementations, it is often better to share the project file system across servers.

Important: The project file system should be monitored to ensure that adequate free space remains. If the project file system runs out of free space, this might corrupt the entire repository, requiring a restore from backup.

Creating a separate project mount on UNIX

It is suggested that a separate file system be created and mounted over the default location for projects, the `$DSHOME/Projects` directory. Mount this directory after installing DataStage but before creating projects.

Note: This file system must be expandable without requiring destruction and re-creation.

To implement a separate file system, archive the contents of $DSHOME/Projects with a utility such as tar, delete the contents of the directory, have the system administrator create and mount the new file system, and restore the contents.

An example set of UNIX commands to accomplish this task is:

```
%etlhost:dsadm /usr/dsadm/Ascential/DataStage >
   /bin/tar -cvf /dev/rmt0 /usr/dsadm/Ascential/DataStage/Projects
%etlhost:dsadm /usr/dsadm/Ascential/DataStage >
   rm /usr/dsadm/Ascential/DataStage/Projects/*
%etlhost:root /usr/dsadm/Ascential/DataStage >
   mkfs /usr/dsadm/Ascential/DataStage/Projects 1024
%etlhost::root /usr/dsadm/Ascential/DataStage >
   mount /usr/dsadm/Ascential/DataStage/Projects
%etlhost::root /usr/dsadm/Ascential/DataStage >
   chgrp dstage /usr/dsadm/Ascential/DataStage/Projects
%etlhost:root /usr/dsadm/Ascential/DataStage >
   chown dsadm /usr/dsadm/Ascential/DataStage/Projects
%etlhost::dsadm /usr/dsadm/Ascential/DataStage >
   /bin/tar -xvf /dev/rmt0 /usr/dsadm/Ascential/DataStage/Projects
```

Creating a project creates a new directory that appears under `$DSHOME/Projects`. After creating a project, a file system can be created to contain that project and be mounted over the `$DSHOME/Projects/ProjectName` directory using the

technique previously described. Creating such file systems is not suggested if `$DSHOME/Projects` is itself a separate file system.

4.34 Verifying project security settings and roles

By default, DataStage project file and directory ownership is set to the user and group of the os-mapped user IS that creates them from within the administrator client. Within administrator, project security can be specified by group membership, InfoSphere Information Server role, or individual.

If you want to set up the system so that it distinguishes between product managers, developers, and operators, set up groups for each class of user. Each user is then allocated to the product manager, developer, or operator secondary group. You can then use the DataStage Administrator client to assign the appropriate DataStage groups to each project.

4.35 Configuring and verifying client installations

Once the DataStage server has been installed, the DataStage client should be installed on each client workstation. This section provides specific requirements and notes regarding the DataStage client.

The version of the DataStage client is tightly tied to the version of the corresponding DataStage Server. The product release notes will detail which client versions are compatible with a particular server.

Important: You must always match the version numbers of the DataStage client and server. This eliminates any potential compatibility issues between the DataStage server and plug-in GUI clients or DataStage clients.

For example, DataStage for Windows 8.1 requires and is only compatible with Version 8.1 of the DataStage client. Furthermore, DataStage client release 8.1 should only be used against a Version 8.1 DataStage server on Windows. (This should not be confused with release 8.1 for UNIX and UNIX System Services platforms.)

For this reason, it is often necessary to install and maintain multiple DataStage client versions on a single workstation. This is particularly the case when developing against different server platforms and when performing a DataStage server upgrade.

4.35.1 DataStage Multi-Client Manager

The DataStage Multi-Client Manager (MCM) allows multiple versions of the DataStage client to be installed on a single workstation. Only one version can be active at any time, but the MCM allows switching between installed versions.

The DataStage Multi-Client Manager is now part of the default client install. If you do not want to installed or configured the MCM on your workstation, select a custom client install. This provides an option to not install the Multi-Client Manager.

Otherwise, the DataStage Multi-Client Manager installs a Windows service. During installation, it prompts you for the user name and password of a Windows administrator to install and start this service. This service needs to be run using a user account that is part of the administrator group on the local machine, because the service must alter settings in the Windows registry. It cannot be the built-in local administrator. Consider the following:

- Even for local administrator accounts, you must supply the fully qualified Windows login name and password. This is of the form DOMAIN\USER, where DOMAIN is the name of your Windows server for local (non-domain) logins.
- Before the DataStage Client installer is run, the user must exist and be in that group. Also, the local security policy must be set up so that the user account has the *logon as a service* user rights assigned to it.
- If a user account that is in the administrators group on the local machine is not specified, then the service will not be placed or installed in the service control manager, and so the MCM will not work.

If a valid user account is specified but does not have the *logon as a service* user rights, then the service will be installed into the service control manager, but it will not be started. The user would have to manually go to the service control manager, re-enter the user account and password for the service, and start the service. At this point the service control manager would automatically assign the correct user rights to the account. Rebooting the machine does not solve the problem.

Note: The Multi-Client Manager installs the *DataStage Multi-Client Manager* Windows service using the administrative user and password that you specify. If you change the password for this account, you must update the service startup properties to use the new password.

4.35.2 WAN development considerations

The DataStage client is a four-tier client/server application. As such, its internal communication protocol includes many messages. Therefore, it is not intended for networks where the latency (roundtrip transmission time for messages) is large.

When using DataStage for remote or distributed development across a wide area network (for example, developers across the world communicating with a central server), it is better to configure a centralized Windows *terminal server* using Microsoft Remote Desktop, CITRIX, VNC, or similar technologies. In these configurations, the DataStage client would be installed on the Windows machine that is co-located with the DataStage Servers.

4.35.3 Secure client installation considerations

Implementing secure installation requires that users other than dsadm be restricted from administrative functions. In addition to InfoSphere Information Server roles, this is accomplished by performing a custom installation and de-selecting the DataStage Administrator client on all workstations other than those authorized to use dsadm. Figure 4-9 depicts a custom client install panel.

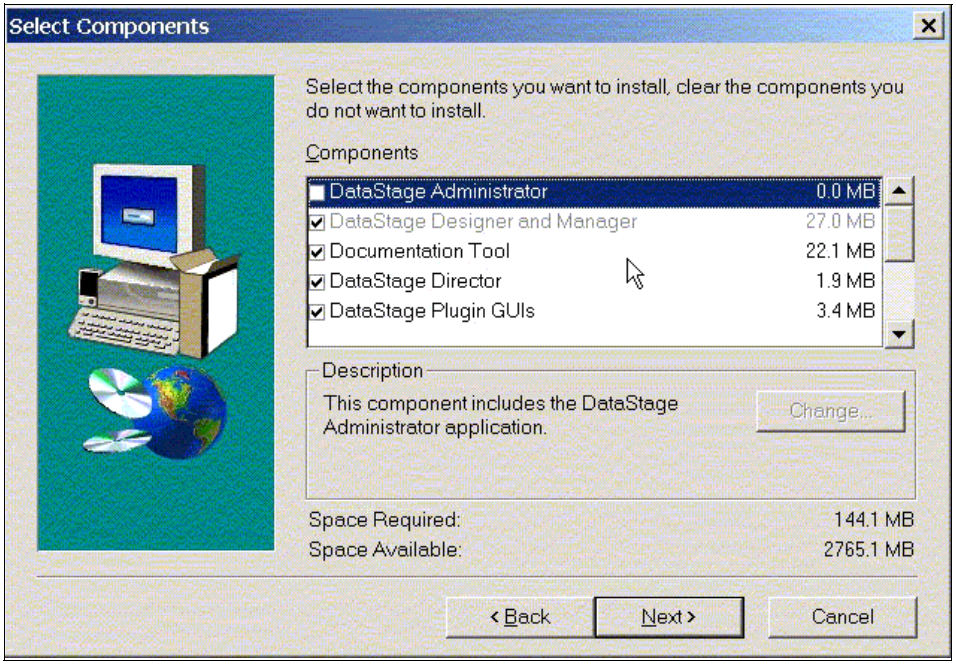


Figure 4-9 Custom client installation

This step is important in securing your installation. The workstations on which the administrator function is installed should be password secured and listed in a client installation inventory.

An effective method of specifying which workstations are to receive which functions is a table such as Table 4-32. We describe our requirements in terms of the business function that the user supports wherever possible. The installer must certify that the administrator function is installed only on the workstations that are specified.

Table 4-32 Workstation administrator functions

Workstation	User	Administrator	Server	Version	Comments
ws127	johnl	Yes	No	Yes	John is the primary project administrator responsible for all areas of the project.
ws324	paulm	Yes	No	Yes	Paul is the secondary project administrator responsible for all areas of the project.
ws718	georgh	No	No	No	
ws817	richards	No	No	No	
ws887	mannie	No	Yes	No	Mannie is a contractor who will develop server routines.
ws888	moe	No	Yes	No	As above.
ws889	jack	No	Yes	No	As above.

4.35.4 Enterprise Application PACKs

If you want to install and run any of the DataStage Enterprise Application PACKs, you must first uninstall the previous version of the DataStage client (if installed). Failure to do so will result in the stage editors for these PACKs failing to load in the Designer client.

Fix packs, patches, and refreshed install packages might be available for InfoSphere Information Server and its components. Periodically check the following web page for the current listings:

http://www.ibm.com/software/data/infosphere/support/info-server/download.html

5

Parallel configuration files

This chapter highlights the IBM InfoSphere DataStage parallel configuration files. A configuration file tells the DataStage Parallel Engine how to use underlying system resources, such as processing, temporary storage, and data set storage. In more advanced environments, it can also define other resources such as databases and buffer storage. At run time, DataStage first reads the configuration file to determine what system resources are allocated to it, and then distributes the job flow across those resources.

When you modify a system by adding or removing nodes or disks, you must modify the configuration files accordingly. Because the Parallel Engine reads a configuration file every time that it runs a job, it automatically scales the application to fit the system without having to alter the job design.

There is not necessarily one ideal configuration file for a system because of the high variability between the way different jobs work and the varying performance for different jobs and applications. For this reason, use multiple configuration files to optimize overall throughput and to match job characteristics to available hardware resources and business needs. At run time, the configuration file is specified through the environment variable `$APT_CONFIG_FILE`.

The information in this section supplements the product documentation. For more information about setting up and maintaining a configuration file, see the *IBM InfoSphere DataStage and QualityStage Designer Client Guide*, SC18-9893-02.

DataStage provides a configuration file editor in the Designer client (Figure 5-1). This graphical tool displays all configuration files located in the default directory (*Configurations* under the DataStage install directory). You can view, edit, and verify created configuration files.

The CHECK option in this dialog box verifies connectivity to the servers and resources listed in the file. It also warns if any operating system limits are too low to handle basic resource allocations for a given configuration file. Because the configuration file editor is not associated with a particular job design, it cannot determine overall resource needs that depend on the size and complexity of a given job.

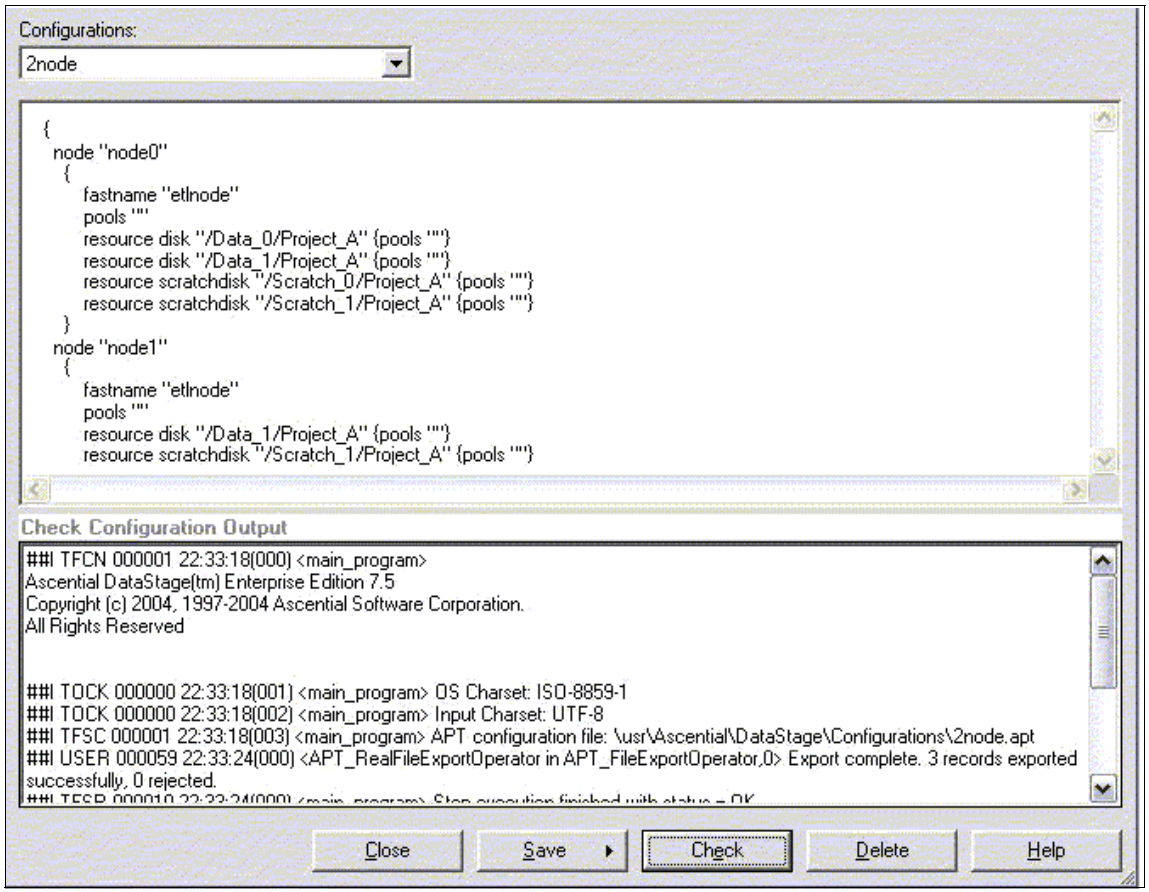


Figure 5-1 Configuration file editor

An alternate way of checking a parallel configuration file is using the **orchadmin** command, which verifies the configuration file pointed to by the environment variable $APT_CONFIG_FILE:

```
orchadmin check
```

The dsadm administrator should use an alias to allow easy access to the configurations directory. Typically, this alias (and others) is added to the login profile of dsadm, for example:

```
A.1.54       alias cdcf="cd
/usr/dsadm/Ascential/DataStage/Configurations"
```

A

Configurations and checklists

This appendix provides sesveral example configurations and checklists to aid in the installation of IBM InfoSphere DataStage in various environments.

Installation and configuration checklist

The checklist in Table A-1 outlines the steps necessary to install and configure DataStage. For information about each step, see the sections later in this appendix. See also the *DataStage Install and Upgrade Guide*, Part No. 00D-018DS60.

Table A-1 Installation and configuration checklist

Complete	Task
	Verify operating system configuration and resource limits.
	Verify RAID/SAN configuration.
	Verify/configure file systems, available space.
	Verify connectivity and network configuration.
	Configure operating system users, groups, and associated permissions.
	Verify C++ compiler and runtime requirements.
	Check product release notes.
	Install DataStage/Parallel Framework.
	Verify install log.
	Install DataStage patches (if applicable).
	Install and configure optional DataStage components.
	Configure post-install operating system settings (if applicable).
	Verify cluster/grid configuration (if applicable).
	Configure and verify DataStage environment and default settings.
	Configure DataStage administrator environment (command line).
	Configure and verify database connectivity.
	Configure and verify ODBC connectivity.
	Create/verify DataStage projects.
	Verify project permissions and security settings/roles.
	Configure and verify client installs.
	Create and verify configuration files.

DataStage administrator UNIX environment

The information summarized in this section includes suggested shortcuts and environment settings for the DataStage super-user account (default dsadm). They are listed in Table A-2, Table A-3, and Table A-4 on page 129.

Table A-2 Environment variables

Environment variable	Setting
$APT_ORCHHOME	Location of PXEngine subdirectory in DataStage install (for example, `/usr/dsadm/Ascential/DataStage/PXEngine`)
$DSHOME	Location of DSEngine subdirectory (for example, `/usr/dsadm/Ascential/DataStage/DSEngine`)
$PATH	`$APT_ORCHHOME/bin:$DSHOME/bin:$PATH`
$LIBPATH *	`$APT_ORCHHOME/lib:$DSHOME/lib:$LIBPATH`
$DSROOT	Root directory of DataStage installation (for example, `/usr/Ascential/DataStage`)
$APT_CONFIG_FILE	`$DSROOT:/Configurations/default.apt`

The actual name of $LIBPATH is platform-dependent, as shown in Table A-3.

Table A-3 Platforms

Platform	Library path environment variable
AIX	LIBPATH
HP-UX (PA-RISC)	SHLIB_PATH
HP-UX (Itanium), LINUX, Solaris	LD_LIBRARY_PATH

The shell shortcuts (aliases) in Table A-4 can be used to move between major DataStage directories.

Table A-4 Aliases

Alias (shell shortcut)	Description	Setting
cdpj	cd to Project dir	`cd /usr/dsadm/Ascential/DataStage/Projects`
cdcf	cd to Config dir	`cd /usr/dsadm/Ascential/DataStage/Configurations`

Alias (shell shortcut)	Description	Setting
cdds	cd to DataStage root	`cd /usr/dadm/Ascential/DataStage`

Installing and configuring multiple server instances

Starting with release 8.1, it is now possible to install several instances of IBM InfoSphere Information Server on a single UNIX or LINUX platform. This means that you can install a new release while maintaining your current release, and install multiple instances of the new release. This allows you, for example, to set up different permissions for different users and effectively limit them to a particular DataStage server instance and associated projects.

Installing or configuring additional InfoSphere Information Server component instances on a single machine works much like a regular installation, but you *must* take special care to not accept any default values for paths or ports. *Every* port used by InfoSphere Information Server will need to be manually configured for each additional instance.

When performing additional installs, be sure to choose the **New Install** and **Custom Install** options during the installer interrogation process to ensure that no ports or paths are re-used. Failure to do so can result in severe damage to the existing installations, possibly rendering them inoperable or unrecoverable.

Prerequisites

Before installing:

- Shut down all InfoSphere Information Server processes on the machine that you will be working on before attempting the install.
- Remove the `/.dshome` file.
- When the install is completed, bring only the new instance online. If everything is working fine, bring up other instances. If anything stops working, most likely a path or port is doing double duty.
- For instances that support the same RDBMS for XMETA, you can use the same RDBMS instance for those instances, if desired.
- Remove all of the auto-start logic from the RC.D directories. The directories/logic used for auto-start varies by platform, so consult the system administrator for your platform.

- If you want to run multiple DSRPC daemons on the same server, you will have to choose a new iTag for each additional instance. You should be prompted for this value if you choose the custom install option. An iTag is a 3-digit hexadecimal value used to identify the DSRPC instance. The default value is ADE. We suggest using a version number such as 810, 801, 811, and so forth.

Finally, as always, take a *full* system backup before attempting this.

Configuring remote DB2

This section provides background information and the steps required to configure connectivity for the WebSphere DataStage DB2 Enterprise Stage.

As a native parallel component, the DB2 Enterprise Stage is designed for maximum performance and scalability. These goals are achieved through tight integration with DB2 Enterprise Server Edition on UNIX, including direct communication with each DB2 database node, and reading from and writing to DB2 in parallel (where appropriate) using the same data partitioning as the referenced tables.

The DB2 Enterprise Stage requires, and provides tight integration with, DB2 Enterprise Server Edition with Data Partitioning Facility (DPF) on UNIX. Both WebSphere DataStage and DB2 Enterprise Server Edition must be running on the same operating system and version.

This section does not highlight configuration requirements for DataStage Stage types that provide connectivity to other DB2 platforms, including:

- DB2 API
- DB2 Load
- Dynamic RDBMS
- ODBC/Enterprise Stages

DB2 Enterprise Stage architecture

This section outlines the high-level architecture of the native parallel DB2 Enterprise stage, providing relevant background to understanding its configuration, as detailed in the remaining sections of this document.

The DataStage Parallel Engine provides a remote DB2 configuration, separating the primary ETL server (conductor node) from the primary DB2 server (coordinator or node zero) using the native parallel DB2 Enterprise Stage. Because DataStage is tightly integrated with the DB2 servers, routing of data to

individual nodes based on DB2 table partitioning, configuration is provided by a combination of DB2 client and DataStage clustered processing.

The primary ETL server (conductor node) must have the 32-bit DB2 client installed and configured to connect to the remote DB2 server instance. This is the same DB2 client that DataStage uses to connect to DB2 databases through the DB2 plug-in stages (DB2 API, DB2 Load, Dynamic RDBMS) for reading, writing, and import of metadata.

The native parallel DB2 Enterprise stage of DataStage uses the DB2 client connection to pre-query the DB2 instance and determine partitioning of the source or target table. This partitioning information is then used to read/write/load data directly from and to the remote DB2 nodes based on the actual table configuration.

This tight integration is provided by routing data within the DataStage engine to DS nodes configured on the DB2 instance servers. This requires a clustered configuration of DataStage, as detailed in the *DataStage Install and Upgrade Guide*, Part No. 00D-018DS60, and in “Cluster or grid configuration” on page 93.

Note: As with any clustered DataStage Enterprise Edition configuration, the engine and libraries must be installed in the same location on all ETL and DB2 servers in the cluster. This is most easily achieved by creating a shared mount point on the remote DataStage and DB2 nodes through NFS or similar directory-sharing methods.

The DB2 client does not have to be installed in the same location on all servers, as long as all locations are included in the $PATH and $LIBPATH environment variable settings.

The actual connectivity scenario for the DB2 Enterprise stage is:

1. The DataStage primary (conductor) node uses DB2 environment variables to determine DB2 instance. If defined, the environment variable `$APT_DB2INSTANCE_HOME` can be used to specify the location on the DataStage conductor server where the remote DB2 server's `db2nodes.cfg` has been copied. The `db2nodes.cfg` file must reside in a subdirectory named sqllib within `$APT_DB2INSTANCE_HOME`.
2. DataStage reads the `db2nodes.cfg` file from the `sqllib` subdirectory of the specified DB2 instance. This file allows DataStage to determine the individual node names of the DB2 instance.
3. DataStage scans the current parallel configuration file (specified by the environment variable `$APT_CONFIG_FILE`) for node names whose fastname

properties match the node names provided in `db2nodes.cfg`. DataStage must find each unique DB2 node name in the configuration file or the job will fail.

4. The DataStage conductor node queries the local DB2 instance via the DB2 client to determine table partitioning information. The results of this query are then used to route data directly from and to the appropriate DB2 nodes.
5. DataStage starts up jobs across all ETL and DB2 nodes in the cluster. This can be easily verified by setting the environment variable `$APT_DUMP_SCORE` to true and examining the corresponding job score entry placed in the job log within DataStage Director.

Prerequisites

Note the following prerequisites:

- DataStage must be installed on all ETL servers and on each DB2 node in the DB2 cluster.
- The hardware and operating system of the ETL server and DB2 nodes must be the same.
- A DB2 32-bit client must be installed on the primary (conductor) DataStage server.

> **TIP:** Use the **`db2level`** command on the ETL server to identify the version of the database.

- The database must be DB2 Enterprise Server Edition with the Data Partitioning Facility (DPF) option installed.

> **TIP:** Use the **`db2level`** command on the DB2 server to identify the version of the database.

- The DB2 database schema to be accessed by DataStage must not have any columns with user defined types (UDTs).

> **TIP:** Use the **`db2 describe table [table-name]`** command on the DB2 client for each table to be accessed to determine if UDTs are in use. Alternatively, examine the DDL for each schema.

Important: Do not attempt to connect the DB2 Enterprise Stage to a remote database by cataloging the remote database in the local instance. If you attempt to use the stage in this way, you might experience data duplication when working with partitioned instances because the node configuration of the local instance might not be the same as the remote instance.

Setting up DB2 connectivity for remote servers

In the following simplified example configuration, two separate AIX servers are configured:

- db2_server as the DB2 database server
- etl_server as the primary DataStage server

Figure A-1 illustrates this.

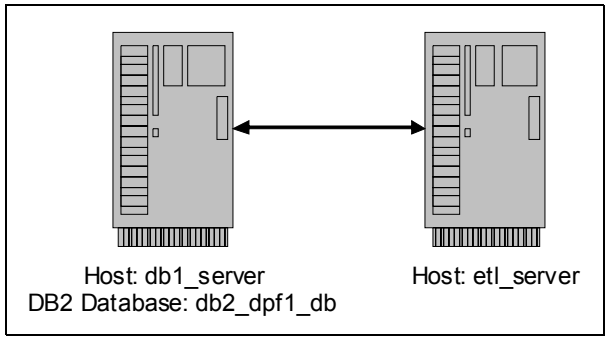


Figure A-1 DataStage DB2 example systems

The steps in this section are demonstrated using the DataStage administrator, which, by default, is `dsadm`. The administration `dsadm` does *not* have to be the local database instance owner.

1. Perform the following actions on *all* members of the cluster *before* installing DataStage on the ETL server:
   a. Create the primary group to which the DataStage users will belong (in this document, this group is the suggested default dstage) and ensure that this group has the same UNIX group ID (for example, 127) on all the systems.
   b. Create DataStage users on all members of the cluster. Make sure that each user has the same user ID (for example, 204) on all the systems, and that every user has the correct group memberships, minimally with dstage as the primary group, and the DB2 group in the list of secondary groups.

c. Add the following users to the DB2 database and ensure that they can log in to DB2 on db2_server. At this step, we are on the DB2 server, and *not* the ETL server. If you fail here, contact your DB2 DBA for support. This is *not* a DataStage issue.

```
/db2home/db2inst1@db2_server>
. /db2home/db2inst1/sqllib/db2profile
/db2home/db2inst1@db2_server>
db2 connect to db2_dpf1_db user dsadm using db2_psword
A.1.55
A.1.56    Database Connection Information
A.1.57
A.1.58 Database server        = DB2/6000 8.2.2
A.1.59 SQL authorization ID   = DSADM
A.1.60 Local database alias   = db2dev1
```

2. Enable the **rsh** command on all servers in the cluster. The simplest way to do this is to create a `.rhosts` file in the home directory of each DataStage user who has the host name or IP address of all members of the cluster, and then setting the permissions on this file to 600. This must be done for each user on all members of the cluster. Modern security systems might prohibit this method, but it will serve as an adequate example of the requirement. Contact the system administrators for the cluster for assistance.

   The commands to be performed on each node of the example system to implement the rhosts method are:

   ```
   echo "etl_server dsadm" > ~/.rhosts
   echo "db2_server dsadm" >> ~/.rhosts
   chmod 600 ~/.rhosts
   ```

   An example of the validation of the etl_server is:

   ```
   /home/dsadm@etl_server>
   rsh db2_server date
   A.1.61 Wed Jan 18 15:40:51 CST 2006
   ```

3. Install a 32-bit DB2 client if one is not installed on the primary ETL server (the server on which DataStage is installed and on which the DS repository resides, also known as the *conductor node*).

   a. Make dsadm the owner of the client. While the software will be installed in `/usr`, management directories and components appear under the home directory of this owner, the top of which is `~/sqllib`. For dsadm on our sample AIX system, this is `/home/dsadm/sqllib`.

   b. Comment out the call to `~/sqllib/db2profile` that the client install puts into the .profile of dsadm. If you do not, DataStage will not operate. It will find DB2 libraries before it finds DataStage libraries.

c. Edit ~/sqllib/db2profile to export INSTHOME, DB2DIR, and DB2INSTANCE.

4. The DB2 DBA must now catalog all the databases that you want to access on the DB2 server into this instance of the DB2 client.

   Ensure that dsadm can log in to DB2 on the db2_server. At this step, we are on the ETL server, and *not* the DB2 server. If you fail here, contact your DB2 DBA for support. This is *not* a DataStage issue.

```
/home/dsadm@etl_server>
 . /home/dsadm/sqllib/db2profile
/home/dsadm@etl_server>
 db2 connect to db2dev1 user dsadm using db2_psword
A.1.62
A.1.63 Database Connection Information
A.1.64
A.1.65 Database server        = DB2/6000 8.2.2
A.1.66 SQL authorization ID   = DSADM
A.1.67 Local database alias   = db2dev1
```

5. Ensure that the remote database is cataloged.

```
A.1.68 home/dsadm@etl_server> db2 "LIST DATABASE DIRECTORY"
A.1.69
A.1.70 Database alias                       = db2dev1
A.1.71 Database name                        = db2_dpf1_db
A.1.72 Node name                            = db2_server
A.1.73 Database release level               = a.00
A.1.74 Comment                              =
A.1.75 Directory entry type                 = Remote
A.1.76 Authentication                       = SERVER
A.1.77 Catalog database partition number    = -1
```

6. Log out of the ETL server and log back in to reset all the environment variables to their original state. Edit $DSHOME/dsenv to include the following information. (The bolded underlined items should be substituted with appropriate values for your configuration.) We assume that the $DB2DIR directory is the same on all nodes in our cluster. This ensures that $PATH and $LIBPATH are correctly set for the remote sessions, as well as the local session, without resorting to individual files on each member of the cluster.

   On operating systems other than AIX (our example system), $LIBPATH might be $SHLIB_PATH or $LD_LIBRARY_PATH.

```
A.1.78 ##############################################
A.1.79 #  DB2 Setup section of dsenv
A.1.80 ##############################################
A.1.81 #DB2DIR is where the DB2 home is located
```

```
A.1.82 DB2DIR=/usr/opt/db2_08_01; export DB2DIR
A.1.83
A.1.84 #DB2INSTANCE is the name of the DB2 client where
A.1.85 #the databases are cataloged
A.1.86 DB2INSTANCE=db2inst1; export DB2INSTANCE
A.1.87
A.1.88 #INSTHOME is the PATH of where the client instance is located
A.1.89 #(usually the home directory of the instance owner.
A.1.90 I NSTHOME=/home/db2inst1; export INSTHOME
A.1.91
A.1.92 #Include the sqllib, bin, adm and misc to the PATH
A.1.93 PATH=$PATH:$DB2DIR/bin; export PATH
A.1.94 THREADS_FLAG=native; export THREADS_FLAG
A.1.9
A.1.96 #Include the DB2 libraries into the LIBPATH AIX
A.1.97 #or LD_LIBRARY_PATH for SUN and Linux
A.1.98 LIBPATH=$LIBPATH:$DB2DIR/lib; export LIBPATH
```

Important: The DataStage libraries *must* be placed *before* the DB2 entries in $LIBPATH ($SHLIB_PATH or $LD_LIBRARY_PATH). DataStage and DB2 use the same library name "librwtool".

7. Copy the `db2nodes.cfg` file from the remote instance to the primary DataStage server. If you create a user on the DataStage server with the same name as the DB2 remote instance owner (for example, db2inst1), then the `db2nodes.cfg` can be placed in that user's "home directory/sqllib" on the DataStage server (unless that user is already the owner of a db2 instance on the ETL server). Otherwise, create a user-defined environment variable APT_DB2INSTANCE_HOME in the DS administrator, add it to a test job, and have it point to the location of the `sqllib` subdirectory where the `db2nodes.cfg` has been placed. Avoid setting this at the project level so that other DB2 jobs that are connecting locally do not pick up this value.

   In our example, the DB2 server has four processing nodes (logical nodes), the instance owner is db2inst1, the `db2nodes.cfg` file on the DB2 server is `/home/db2inst1/sqllib/db2nodes.cfg`, and this file has the following contents:

   ```
   A.1.99 0 db2_server 0
   A.1.100 1 db2_server 1
   A.1.101 2 db2_server 2
   A.1.102 3 db2_server 3
   ```

   In our example, the ETL server client is owned by dsadm, the APT_DB2INSTANCE_HOME environment variable has been set to

"/home/dsadm/remote_db2config", and this file was copied to `/home/dsadm/remote_db2config/sqllib/db2nodes.cfg` on the ETL server.

8. Ensure that dsadm can connect to the instance using the values in $DSHOME/dsenv instead of ~/sqllib/db2profile. Log out of the ETL server and log back in to reset all the environment variables to their original state:

```
/home/dsadm@etl_server> cd `cat /.dshome`/dsenv
/home/dsadm@etl_server> . ./dsenv
/home/dsadm@etl_server> db2 connect to db2dev1 user dsadm using
db2_psword
```

The following is the database connection information:

```
Database server        = DB2/6000 8.2.2
SQL authorization ID   = DSADM
Local database alias   = db2dev1
```

> **Note:** On an iTag install (see “Installing and configuring multiple server instances” on page 130), the `/.dshome` file might not exist, or it might point to the wrong DataStage instance. In this case, set $DSHOME manually, then use the **cd** (change directory) command to change to that directory.

9. Implement a DataStage cluster (refer to the *Install and Upgrade Guide*, Part No. 00D-018DS60, for more details). In this example, `/etl/Ascential` is the file system that contains the DataStage software system, and it is NFS-exported from the ETL server to the DB2 server, and NFS-mounted exactly on `/etl/Ascential`, a file system owned by dsadm on the DB2 server.
10. Verify that the DB2 operator library has been properly configured by making sure that the link *orchdb2op* exists in the `$APT_ORCHHOME/lib` directory. Normally, this link is configured on install, but if it does not exist, you must run the script `$APT_ORCHHOME/install/install.liborchdb2op`. You will be prompted to specify DB2 Version 7 or 8 (in our case, Version 8).
11. The `db2setup.sh` script located in `$PXHOME/bin/` can run without reporting errors even if they occur, and if there are errors, DataStage will not be able to connect to the databases. Run the following commands and ensure that no errors occur:

```
/home/dsadm@etl_server> db2 connect reset
/home/dsadm@etl_server> db2 connect terminate
/home/dsadm@etl_server> db2 connect to db2dev1 user dsadm using
db2_psword
/home/dsadm@etl_server>
db2 bind ${APT_ORCHHOME}/bin/db2esql.bnd blocking all grant public
/home/dsadm@etl_server> cd ${INSTHOME}/sqllib/bnd
/home/dsadm@etl_server>
```

```
db2 bind @db2bind.lst datetime ISO blocking all grant public
/home/dsadm@etl_server> db2 bind @db2cli.lst datetime ISO blocking
all grant public
```

Note: Datetime ISO currently prevents this bind from succeeding. Omit this option when issuing the bind until this issue has been resolved by development.

```
/home/dsadm@etl_server> db2 connect reset
/home/dsadm@etl_server> db2 connect terminate
/home/dsadm@etl_server> db2 connect to db2dev1 user dsadm using
db2_psword
/home/dsadm@etl_server>
db2 grant bind, execute on package dsadm.db2.esql to group dstage
/home/dsadm@etl_server> db2 connect reset
/home/dsadm@etl_server> db2 connect terminate
```

12. The `db2grant.sh` script located in `$PXHOME/bin/` can run without reporting errors even if they occurred. If there are errors, DataStage will not operate correctly. Run the following commands and ensure that no errors occur. Grant bind and execute privileges to every member of the primary DataStage group (in our case dstage).

```
/home/dsadm@etl_server> db2 connect to db2dev1 user dsadm using
dsadm_db2_psword
/home/dsadm@etl_server>
db2 grant bind, execute on package dsadm.db2.esql to group dstage
/home/dsadm@etl_server> db2 connect reset
/home/dsadm@etl_server> db2 connect terminate
```

13. Create a clustered DataStage configuration file that includes nodes to be used for ETL processing and also includes one node entry for each server in the remote DB2 instance.

    Unless ETL processing is to be performed on the remote DB2 server nodes, the entries `pools ""` should be removed from the default node pool. Each node in the DB2 instance should be part of the same node pool (for example, pools "db2"). Figure A-1 on page 134 shows an example configuration file.

Example A-1 Configuration file

```
{
        node "node1"
        {
                fastname "etl_server "
                pools ""
                resource disk "/worknode1/datasets" {pools ""}
```

```
                resource scratchdisk "/worknode1/scratch" {pools ""}
        }
        node "db2node1"
        {
                fastname "db2_server"
                pools "db2"
                resource disk "/worknode/datasets" {pools ""}
                resource scratchdisk "/worknode/scratch" {pools ""}
        }
}
```

14. Restart the DataStage server.
15. Test server connectivity by trying to import a table definition within DataStage Designer (or DataStage Manager) using the DB2 API plug-in (Server plug-in). If this fails, you do not have connectivity to the DB2 server and need to revisit all the previous steps until this succeeds.

    If this succeeds, check the imported table definitions to be sure that the data types are legitimate.
16. Create a user-defined variable $APT_DB2INSTANCE_HOME in the DataStage project using the administrator client for use in jobs that access DB2. Avoid setting this at the project level so that other DB2 jobs that are connecting locally do not inherit this value. Set this variable in each job to the location of the `sqllib/db2nodes.cfg` file (in our case, `/home/dsadm/remote_db2config`).

Configuring multiple DB2 instances in one job

Although it is not officially supported, it is possible to connect to more than one DB2 instance within a single job. Your job must meet one of the following configurations. (The use of the word *stream* refers to a contiguous flow of data from one stage to another within a single job.)

- Single stream: Two instances only

  Reading from one instance and writing to another instance with no other DB2 instances. (It has not been determined how many stages for these two instances can be added to the canvas for this configuration for lookups.)
- Two streams: One instance per steam

  Reading from instance A and writing to instance A and reading from instance B and writing to instance B.
- Multiple streams with N DB2 sources with no DB2 targets

  Reading from one to n DB2 instances in separate source stages with no other downstream DB2 stages.

To get this configuration to work correctly, you must adhere to all of the directions specified for connecting to a remote instance and the following:

- You must not set the APT_DB2INSTANCE_HOME environment variable. Once this variable is set, DataStage will try to use it for each of the connections in the job. Because a `db2nodes.cfg` file can only contain information for one instance, this creates problems.
- For DataStage to locate the `db2nodes.cfg` file for each of the separate DB2 stages, you must build a user on the DataStage server with the same name as the instances to which you are trying to connect. DataStage's default logic assumes that the instance corresponds to a UNIX user and that the `db2nodes.cfg` file will exist in a `sqllib` subdirectory in that ID's home directory. Therefore, create a `sqllib` subdirectory for each remote instance and place the remote instance's `db2nodes.cfg` there. Because the APT_DB2INSTANCE_HOME is not set, DataStage defaults to these directories to find the config file for the remote instance.

Troubleshooting

If you experience problems, consider the following troubleshooting approaches:

- If you get an error while performing the binds and grants, make sure that user dsadm has privileges to create schema, select on the sysibm.dummy1 table, and bind packages (see installation documentation for the DB2 grants necessary to run the scripts).
- There are several errors while trying to view data from the DB2 Enterprise plug-in that do not represent the actual issue:
  - If you log into DataStage with a user name (for example, dsadm) and try to view data with a different user in the plug-in (user name and password inside of the plug-in), you might get a failed connection. This is because the user name and password inside of the stage are only used to create a connection to DB2 via the client, and then the job runs using the DataStage user (the user name is used to log into DataStage either from the Designer or the Director).
  - The user does not have permission to read the catalog tables
- The user ID used to access the DB2 remote servers has to be set in each of the servers. For example, the dsadm user has to be set up as a UNIX user in the ETL server and in all of the DB2 nodes. Also, make sure that the groups are set correctly because the `db2grant.sh` script only grants permission to the group (in our example, dstage or as an example bd2group).
- The DB2 instance is a service that needs to be running before you can connect to any of the cataloged databases.

- The permission on the resource disk or scratch must be set correctly (mainly for performing a load). When using the load, make sure that the resource disk and scratch are open to dstage, as well as the DB2 instance owner where the data is going to be loaded. Usually, the groups are different, so the permission needs to be set to 777.

Performance notes

In some cases, when using user-defined SQL without partitioning against large volumes of DB2 data, the overhead of routing information through a remote DB2 coordinator might be significant. In these instances, it might be beneficial to have the DB2 DBA configure separate DB2 coordinator nodes (no local data) on each ETL server (in clustered ETL configurations). In this configuration, DB2 Enterprise stage should not include the client instance name property, forcing the DB2 Enterprise operators on each ETL server to communicate directly with their local DB2 coordinator.

Summary of settings

The DB2 libraries must come after the DataStage libraries because both products have libraries with identical names. The DB2 client alters the .profile of the DB2 owner, and this must be removed or DataStage will not function.

The last four lines of the .profile for user dsadm on the ETL server are:

```
A.1.103 home/dsadm @ etl_server >> tail -4 .profile
A.1.104
   A.1.105 # The following three lines were added by UDB and removed by
   IBM IIS.
   A.1.106 # if [ -f /home/dsadm/sqllib/db2profile ]; then
   A.1.107 #     . /home/dsadm/sqllib/db2profile
   A.1.108 # fi
```

Environment variables set by /home/dsadm/sqllib/db2profile must be supplied after the native DataStage environment variables. This is done with the `dsenv` file for the DataStage server.

The last lines of the `dsenv` file with DB2 setup information added are:

```
/etl/Ascential/DataStage/DSEngine @ etl_server >> tail -8 dsenv
A.1.109
A.1.110 # DB2 setup section
A.1.111 DB2DIR=/usr/opt/db2_08_01; export DB2DIR
A.1.112 DB2INSTANCE=dsadm; export DB2INSTANCE
A.1.113 INSTHOME=/home/dsadm; export INSTHOME
```

```
A.1.114 PATH=$PATH:$DB2DIR/bin; export PATH
A.1.115 THREADS_FLAG=native; export THREADS_FLAG
A.1.116 LIBPATH=$LIBPATH:$DB2DIR/lib; export LIBPATH
```

The contents of the `db2nodes.cfg` file located in `/home/dsadm/remote_db2config/sqllib` are:

```
A.1.117 /home/dsadm/remote_db2config/sqllib @ etl_server >> cat
db2nodes.cfg
A.1.118
A.1.119 0 db2_server 0
A.1.120 1 db2_server 1
```

Increasing DataStage Server Edition memory on AIX

When running DataStage Server jobs on AIX platforms, it might be necessary to increase the available memory usage to process very large numbers of records in memory. This appendix details the process of increasing this memory allocation and outlines a process for ensuring that these changes will not adversely affect DataStage parallel jobs running on the same environment.

AIX implements a segmented shared memory model. The environment variable settings and changes to the DataStage Server engine configuration increase the amount of shared memory available to DataStage Server jobs.

To increase available DataStage Server memory usage on AIX:

1. Change the following settings in the `uvconfig` file located in the home directory (`DSEngine`) of DataStage. These setting change the shared memory address points. (Edit the file as user dsadm.)

   ```
   DMEMOFF 0x90000000
   PMEMOFF 0xa0000000
   ```

2. While logged in as user dsadm, apply these changes to the DataStage Server engine by running the following UNIX commands:

   ```
   cd DSEngine
   . ./dsenv
   bin/uv -admin -stop
   bin/uv - admin -regen
   bin/uv -admin - start
   ```

For each DataStage Server job, add the following job-level environment variable and setting. (The equal sign is part of the environment variable value and is required.)

```
LDR_CNTRL = MAXDATA=0x30000000
```

Important: The LDR_CNTRL setting must only be applied to Server Edition jobs. Setting this value for parallel jobs causes these jobs to fail.

Using HP-UX 11 memory on Windows

Memory windows allow applications to use more than the 1.75 GB limit imposed on 32-bit processes by HP-UX 11 and 11i. To use this feature for DataStage parallel jobs, one first needs to understand exactly how much memory is being used by the job. For parallel jobs, this can be estimated by calculating the total data size of all lookup tables used by a particular job.

Let us take the example of a 32-processor machine with 32 GB of memory, called hptest. The default configuration file for this machine would contain 16 nodes, each with the fast name entry *hptest*. If the jobs that are going to be run on this will use a maximum of 8 GB, we would want to use at least eight memory windows, as each window can hold up to 1 GB of memory. Below are the instructions for how to set up the system and the parallel engine to use eight memory windows.

1. Change the kernel tunable max_mem_window from 0 to 40 and reboot the machine.
2. Create seven new entries in the `/etc/hosts` file, all pointing to the current machine. For example, if the machine is called hptest, the original `/etc/hosts` file probably looks similar to the following:

```
A.1.121 # @(#)hosts $Revision: 1.9.214.1 $ $Date: 96/10/08 13:20:01 $
A.1.122 #
A.1.123 # The form for each entry is:
A.1.124 # <internet address>    <official hostname> <aliases>
A.1.125 #
A.1.126 # For example:
A.1.127 # 192.1.2.34    hpfcrm  loghost
A.1.128 #
A.1.129 # See the hosts(4) manual page for more information.
A.1.130 # Note: The entries cannot be preceded by a space.
A.1.131 #       The format described in this file is the correct
format.
A.1.132 #       The original Berkeley manual page contains an error in
```

```
A.1.133 #       the format description.
A.1.134 #
A.1.135
A.1.136 134.168.56.29   hptest
Add the following line:
A.1.137 134.168.56.29   hptest1 hptest2 hptest3 hptest4 hptest5
hptest6 hptest7
A.1.138
```

3. If your machine is set up to trust a limited number of machines, you will need to add each of the new host names (hptest1, … , hptest7) to your ~/.rhosts or /ext/hosts.equiv file.

4. Create an entry in the /etc/services.window for each host name alias being used. For example:

```
A.1.139 #
A.1.140 # /etc/services.window
A.1.141 #
A.1.142 # The format of this file is name followed by a space/tab
followed
A.1.143 # by a unique user key. A memory window application uses the
A.1.144 # getmemwindow(1M) command to extract the user key and then
passes
A.1.145 # that key to the setmemwindow(1M) command. Using the same
key
A.1.146 # places those applications in the same memory window.
A.1.147 #
A.1.148 #       For example:
A.1.149 #
A.1.150 #       winid=getmemwindow database1
A.1.151 #       setmemwindow -i $winid startup_script arg1 arg2
arg3.
A.1.152 #
A.1.153 hptest 10
A.1.154 hptest1 11
A.1.155 hptest2 12
A.1.156 hptest3 13
A.1.157 hptest4 14
A.1.158 hptest5 15
A.1.159 hptest6 16
A.1.160 hptest7 17
```

5. Once you have done this, modify your 16-node $APT_CONFIG_FILE, so the fastname entries are divided evenly across hptest through hptest7.

6. Ensure that the user can run **remsh** to the current node. For example, try the following command:

   ```
   remsh hptest ls
   ```

7. Add the following file (named `startup.apt`) in the `$APT_ORCHHOME/etc` directory and make sure that it is executable. If at any time you want to disable the use of memory windows in the future, you can do so by setting the $APT_NO_STARTUP_SCRIPT environment variable.

   ```
   A.1.161 #!/bin/sh
   A.1.162 shift 2
   A.1.163 winid=$(getmemwindow $8)
   A.1.164 echo "hostname="$8 "windowid="$winid
   A.1.165 setmemwindow -i $winid -p $$
   A.1.166 exec $*
   ```

8. Test this new configuration with a sample parallel job and the new $APT_CONFIG_FILE. (A simple column generator to peek will do).

Note: When using the memory windowing technique for large lookup tables, you must hash partition the incoming data and all reference tables using the same key columns. The default “Entire” partitioning will not use memory windowing.

Estimating the size of a parallel data set

For the advanced user, this section provides a more accurate and detailed way to estimate the size of a parallel data set based on the internal storage requirements for each data type, as listed in Table A-5.

Table A-5 Data type sizes

Data type	Size
Integers	4 bytes
Small integer	2 bytes
Tiny integer	1 byte
Big integer	8 bytes
Decimal	(precision+1)/2, rounded up
Float	8 bytes

Data type	Size
VarChar(n)	n + 4 bytes for non-NLS data 2n + 4 bytes for NLS data (internally stored as UTF-16)
Char(n)	n bytes for non-NLS data 2n bytes for NLS data
Time	4 bytes 8 bytes with microsecond resolution
Date	4 bytes
Timestamp	8 bytes 12 bytes with microsecond resolution

For the overall record width:

```
add (# nullable fields)/8 for null indicators
one byte per column for field alignment (worst case is 3.5 bytes per
field)
```

Using the internal DataStage C++ libraries, the method `APT_Record::estimateFinalOutputSize()` can give you an estimate for a given record schema, as can `APT_Transfer::getTransferBufferSize()`, if you have a transfer that transfers all fields from input to output.

Windows XP Service Pack 2 firewall configuration

Starting with Service Pack 2, Microsoft has introduced the Windows Firewall—a replacement for the internet connection firewall (ICF) in the previous version of Windows XP. Windows Firewall is designed to discard unsolicited incoming traffic, providing a level of protection for computers against malicious users or programs. Starting with SP2, the Firewall is enabled on all network connections by default. This new behavior impairs server communications when the server is hosted on XP computers. In this section we describe how to configure the Windows Firewall to work with server components of the IBM Information Integration Solutions suite.

Disclaimer: Windows XP is not intended to support server applications, but it can be configured to do so. IBM does not officially support running many of the DataStage Server components on the Windows XP platform. XP is only a supported platform for the DataStage client. If you are running an IBM Information Integration server component on Windows XP, plan your move to a supported platform.

To begin, execute the following steps:

1. Determine whether you have Windows XP Service Pack 2 installed. From the Windows desktop, right-click **My Computer**, and then select **Properties**.
2. Select the **General** tab and look under the System heading.

   There are three possible scenarios, depending on or your service pack level, as listed in Table A-6.

Table A-6 Service pack and action

Appears on Properties bar	Action
Windows XP and Service Pack 2 and later	Move to step 3.
Windows XP Only	Forward this document to your IS group so that they are aware of the implications when they roll out SP2.
Windows 2000, Server 2003	Do not worry if you are not running Windows XP and thus do not have the new Windows Firewall installed.

3. Choose a method for configuring the firewall (Table A-7).

Table A-7 Configuration methods

Method	Notes
Disable the firewall entirely.	Easiest but least secure in an open environment. Your corporate firewall might already protect your computer.
Open the firewall on a per-application basis.	Easy, and more secure than total disablement. Possibly redundant if you sit behind a corporate firewall.
Open the firewall on a per-port basis.	The most traditional method of working with a firewall. Possibly redundant if you sit behind a corporate firewall.

4. Open the Windows Firewall Configuration page (Figure A-2).

> **Note:** No matter which method you choose you will have to open the firewall configurations screen. You can find it by clicking **Start** ∅ **Settings** ∅ **Control Panel** ∅ **Windows Firewall**.

Consult the table in step 3 and then choose either step 5, 6, or 7. You only need to perform one of the steps.

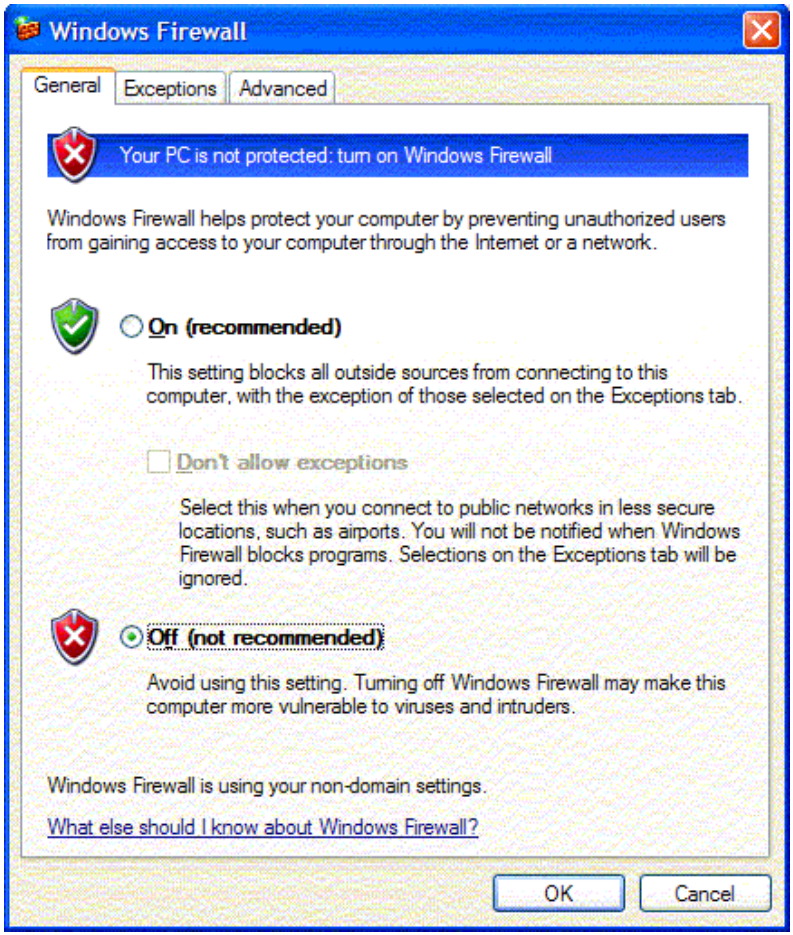


Figure A-2 Firewall configuration

5. Disable all firewall functionality. Once you have the firewall configuration window open select **Off** and then choose **OK**.

6. Disable the firewall per IBM application. Once you have the firewall configuration window open, make sure that the Do not allow exceptions box is *not* selected, then click the **Exceptions** tab (Figure A-3).

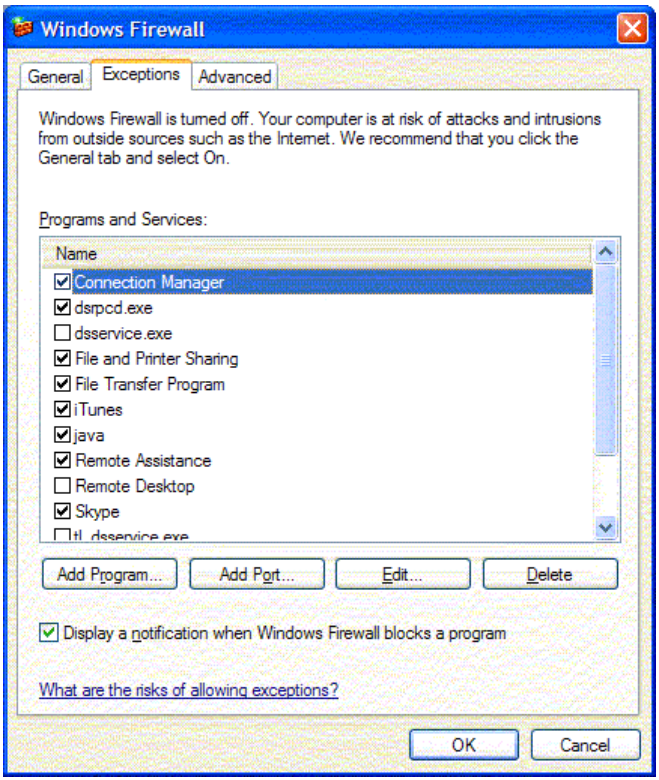


Figure A-3 Firewall exceptions

7. Use the Add Programs button to add the appropriate IBM InfoSphere Information Server applications (Figure A-4).

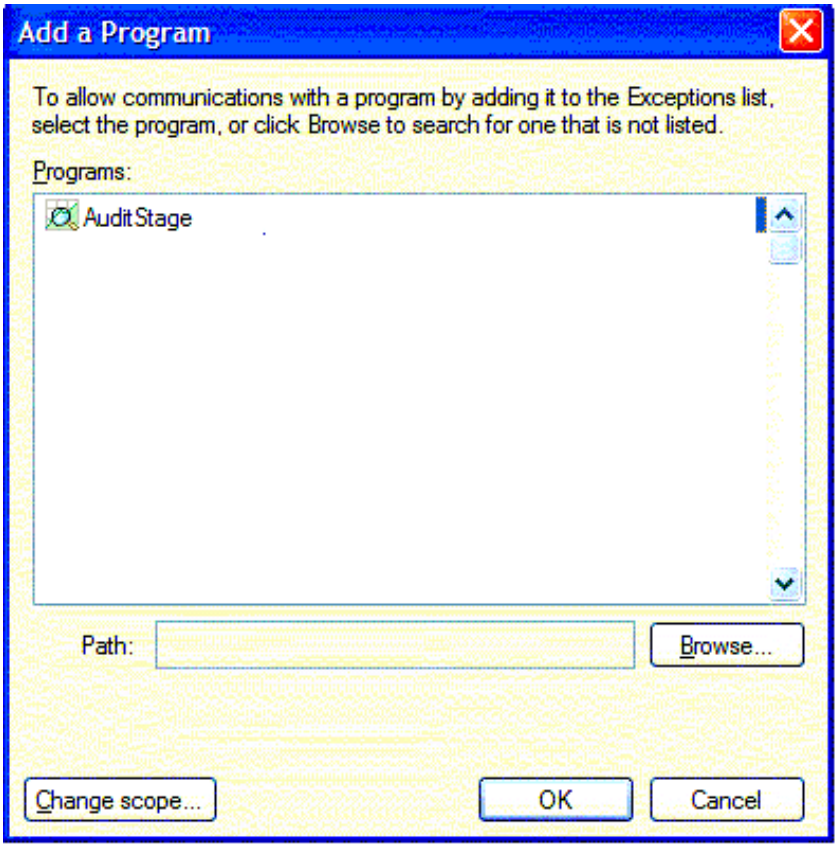


Figure A-4 Add applications

You might need to browse to the location of the IBM application. A table of all IBM application names can be found in the N-Network Ports used section. The exact location depends on the directory location that you selected when installing the products. You can change the scope of the network use of the opened application. Access can be limited to certain networks or IP addresses. The scope options are the same for application or port method.

8. Enable only certain ports on the firewall.

   Once you have the Firewall Configuration panel open (Figure A-2 on page 149), click the **Allow exceptions** box, then click the **Exceptions** tab.

   Use the **Add Ports** button to add the appropriate IBM port applications.

More information

For more information about Windows XP SP2, consult the following resources:

- New Networking Features in Windows XP Service Pack 2 (the January 2004 Cable Guy article)

  http://technet.microsoft.com/en-us/library/bb877964.aspx

- Changes to Functionality in Microsoft Windows XP Service Pack 2

  http://www.microsoft.com/downloads/en/details.aspx?FamilyID=7bd948d7-b791-40b6-8364-685b84158c78&DisplayLang=en

- Deploying Windows Firewall Settings for Microsoft Windows XP with Service Pack 2

  http://www.microsoft.com/downloads/en/details.aspx?FamilyID=4454e0e1-61fa-447a-bdcd-499f73a637d1&displaylang=en

- Troubleshooting Windows Firewall in Microsoft Windows XP Service Pack 2

  http://www.microsoft.com/downloads/en/details.aspx?familyid=a7628646-131d-4617-bf68-f0532d8db131&displaylang=en

DataStage ports used in Windows platforms

This section lists all DataStage ports. You can change the scope of the network use of the opened ports. Access can be limited to certain networks or IP addresses. The scope options are the same for application and port method and are covered in “Defining the scope for a program or port” on page 154. Table A-8 summarizes the DataStage ports.

Table A-8 DataStage ports

Component/protocol	Default port number	Configurable?	Information server tier
IBM DB2 database for the metadata repository (default)	50000	Yes	Services
IBM DB2 database for the analysis database (default)	50000	Yes	Services, engine, client
DHTML reports server	16581	Yes	Client
IBM Information Server web-based clients	9080	Yes	Services, engine, client
IBM Information Server web-based clients - HTTPS	9443[a]	Yes	Client

Component/protocol	Default port number	Configurable?	Information server tier
WebSphere Application Server administrative console (redirects to HTTPS)	9060[b]	Yes	Client
WebSphere Application Server administrative console (HTTPS)	9043	Yes	Client
IBM Information Server services (RMI/IIOP)	2809, 9100, 9401-9403	Yes	Services, engine, client
IBM WebSphere Information Services Director services with JMS bindings[c]	7276, 7286, 5558, 5578	Yes	

a. Used only when using HTTPS to access web clients.
b. Used only if you need access to the WebSphere Application Server administrative console.
c. Used only when publishing services using a JMS binding. The port needs to be accessible to services consumers.

Table A-9 lists the engine tier ports.

Table A-9 Engine tier ports

Component/protocol	Default port number	Configurable?	Information server tier
IBM InfoSphere Information Server ASB agent	31531, and a random port number greater than 1024[a]	Yes	Services
IBM InfoSphere Information Server logging agent	31533	Yes	Engine
IBM WebSphere DataStage and QualityStage services	31538	Yes	Engine, client
Parallel job monitors	13400 (port 1) and 13401 (port 2)	Yes	Engine[b]
Parallel engine (APT_PM_STARTUP_PORT)	Multiple ports, uses a port number of 10000 or greater	Yes	Engine
Parallel engine remote process startup (rsh/ssh, multiple nodes only)	22514		Engine
Parallel engine (APT_PLAYER_CONNECTION_PORT, multiple nodes only)	Multiple ports, uses a port number of 11000 or greater	Yes	Engine

a. Can be fixed to a specific port by specifying agent.objectport=# in the `C:\IBM\InformationServer\ASBNode\conf\agent.properties` file after you complete the installation. After designating a specific port, restart the logging agent and the ASB agent so that the change takes effect.
b. Access to port 1 is required only from the conductor node. Access to port 2 is required from the conductor node and the node where the IBM WebSphere DataStage and QualityStage engine is installed, if that node is not the same as the conductor node.

Table A-10 lists the miscellaneous ports.

Table A-10 Miscellaneous ports

Component/protocol	Default port number	Configurable?	Information server tier
FTP	Any port number	Yes	Engine
SFTP	Any port number	Yes	Engine
SSH	Any port number	Yes	Engine
Telnet	Any port number	Yes	Engine
SMTP	Any port number	Yes	Engine
ODBC	Any port number	Yes	Engine
JDBC	Any port number	Yes	Engine
CLI (database platform specific)	Any port number	Yes	Engine
BW Pack	3200-3615	Yes	Engine

Defining the scope for a program or port

You have three options when defining the scope for a program or a port:

- Any computer (including those on the internet)

  Excepted traffic is allowed from any IPv4 address. This setting might make your computer vulnerable to attacks from malicious users or programs on the internet.

- My network (subnet) only

  Excepted traffic is allowed only from an IPv4 address that matches the local network segment (subnet) to which the network connection that received the traffic is attached. For example, if the network connection is configured with an IPv4 address of 192.168.0.99 with a subnet mask of 255.255.0.0, excepted traffic is only allowed from IPv4 addresses in the range 192.168.0.1 to 192.168.255.254.

The My network (subnet) only scope is useful when you want to allow access to a program or service for the computers on a local home network that are all connected to the same subnet, but not to potentially malicious internet users.

- Custom list

  You can specify one or more IPv4 addresses or IPv4 address ranges separated by commas. IPv4 address ranges typically correspond to subnets. For IPv4 addresses, type the IPv4 address in dotted decimal notation. For IPv4 address ranges, you can specify the range using a dotted decimal subnet mask or a prefix length.

  When you use a dotted decimal subnet mask, you can specify the range as an IPv4 network ID (such as 10.47.81.0/255.255.255.0) or by using an IPv4 address within the range (such as 10.47.81.231/255.255.255.0).

  When you use a network prefix length, you can specify the range as an IPv4 network ID (such as 10.47.81.0/24) or by using an IPv4 address within the range (such as 10.47.81.231/24).

  The following is an example custom list:

  ```
  10.91.12.56,10.7.14.9/255.255.255.0,10.116.45.0/255.255.255.0,172.16
  .31.11/24, 172.16.111.0/24
  ```

  You cannot specify a custom list for IPv6 traffic. Once the program or port is added, it is disabled by default in the Programs and Services list.

All of the programs or services enabled from the Exceptions tab are enabled for all of the connections that are selected on the Advanced tab.

Pre-installation checklist

Table A-11 outlines the areas that you must review and the steps that you must complete before you install IBM InfoSphere Information Server. For details about each step, see this paper, the release notes, and the *IBM Information Server Planning, Installation, and Configuration Guide*, GC19-1048-07.

Table A-11 Pre-installation checklist

Complete	Task
	1) Review release notes (InfoSphere Information Server, WebSphere Application Server, DB2).
	2) Review *IBM Information Server Planning, Installation and Configuration Guide*, GC19-1048-07.

Complete	Task
	3) If migrating from previous versions of DataStage or QualityStage, review IBM publication *Migrating to IBM Information Server Version 8,* SC18-9924.
	4) Choose and validate installation architecture/topology.
	5) Validate system requirements for all tiers (engine, domain, repository, client, documentation).
	6) Verify domain (WebSphere Application Server) requirements.
	7) Verify database requirements for metadata repository.
	8) If applicable: Verify database requirements for Information Analyzer analysis database.
	9) Verify and configure disks, volume groups, and file systems.
	10) Verify and configure operating system and resource limits.
	11) Verify connectivity and network configuration.
	12) Configure operating system users, groups, and associated permissions.
	13) Verify and install C++ compiler or runtime libraries, or both.
	14) Verify InfoSphere Information Server Connector requirements.
	15) Download and install fix pack packages (InfoSvr, WebSphere, DB2).
	16) Perform a complete system backup.

Installation and configuration checklist

The checklist in Table A-12 outlines the steps to install and configure IBM InfoSphere Information Server. For details about each step, see the release notes and the *IBM Information Server Planning, Installation, and Configuration Guide*, GC19-1048-07.

Table A-12 Installation and configuration checklist

Complete	Task
	1) Complete all items on the Pre-Install Checklist.
	2) (If migrating from earlier DataStage or QualityStage) Complete the pre-installation migration tasks.

Complete	Task
	3) Create and configure the metadata repository ("xmeta") (if not using supplied DB2).
	4) Create and configure the analysis ("iadb") database (required for Information Analyzer).
	5) Install and configure the domain (WebSphere AS) server (if not using the supplied WebSphere Application Server).
	6) Install IBM InfoSphere Information Server.
	7) Review all install logs.
	8) (If applicable) Install IBM InfoSphere Information Server fix packs and patches.
	9) Review fix pack and patch installer log files.
	10) Install and configure optional IBM InfoSphere Information Server components.
	11) (If applicable) Configure post-install operating system settings.
	12) (If applicable) Verify the cluster/grid configuration.
	13) Configure and verify the InfoSphere Information Server environment and defaults.
	14) Verify the InfoSphere Information Server administration (shell) environment.
	15) Configure information server users, roles, and permissions.
	16) Configure and verify InfoSphere Information Server Connectors.
	17) Configure and verify client installs.
	18) (Where applicable) install client fix packs, patches, and optional components.
	19) Review client install, client fix pack, and client patch install log files.
	20) Create and verify parallel configuration files.
	21) Create and verify InfoSphere Information Server projects.
	22) Verify project permissions and security settings and roles.

InfoSphere Information Server installation settings

The information in Table A-13 summarizes the various settings and values supplied to the IBM Information Server installer, in the order that they are supplied to the installer. Not all options in Table A-13 are displayed. Whether an option is displayed depends on the options that are chosen in the installer.

Table A-13 InfoSphere Information Server installation settings

Configuration topic	Installation option	Default value	Install value
	Installation response file	`/root/is_install.rsp`	
	Installation directory	`/opt/IBM/Information Server/`	
Components	Engine	True	
	Domain	True	
	Metadata repository	True	
	Documentation	True	
License file	Name and location	`root/license.xml`	
Product module and component selection (Depends on license)	Metadata server	True	
	Business glossary	True	
	Information analyzer	True	
	Federation server	True	False
	DataStage and QualityStage	True	
	WebSphere Information Services director	True	
Install type	Typical or custom	Typical	
DataStage server	Install new or upgrade	Install new	

Configuration topic	Installation option	Default value	Install value
Metadata server database connection	Type (DB2 v8, DB2 v9, Oracle 10, SQLServer)	DB2 v9	
	Host	Localhost	
	TCP-IP port	50000	
	Database name	xmeta	
	Database owner	xmeta	
	Password		xmeta1
WebSphere Application Server	Install type: New or configure existing	New	
	WebSphere Application Server profile (for existing install)	default	
	Destination directory	`/opt/IBM/WebSphere/AppServer`	
Metadata server user registry	Internal versus local OS registry	Internal registry	
	AppSvr administrator user		wasadmin
	AppSvr administrator pass		
	Information Server Suite administrator user		isadmin
	Information Server Suite administrator password		
DataStage projects	Project name and path		
Information Analyzer database connection	Type (DB2 v8, DB2 v9, Oracle 10, SQLServer)	DB2 v9	
	Host	Localhost	
	TCP-IP Port	50000	
	Database name	IADB	
	Database owner	iauser	
	Password		

Configuration topic	Installation option	Default value	Install value
DB2 configuration	Language selection	(English always installed)	
	Install location	/opt/IBM/db2/V9	
	Admin server user	dasusr1	
	Admin server password		
	Admin server group	dasadm1	
	Admin server home	/home/dasusr1	
	Instance owner user	db2inst1	
	Instance owner password		
	Instance owner group	db2iadm1	
	Instance owner home	/home/db2inst1	
	Instance port	50000	
	Fenced user	db2fenc1	
	Fenced password		
	Fenced group	db2fadm1	
	Fenced home	/home/db2fenc1	
DataStage configuration	DataStage administrator	dsadm	
	DataStage instance tag	ade	
	DataStage RPC port	31538	
	Install NLS for DataStage	FALSE	TRUE
	MQ plug-in enabled	FALSE	
	MQ plug-in type	Server	
	Oracle operator enabled	FALSE	
	Oracle operator version	10g	
	Legacy SAS operator enabled	FALSE	
	Legacy SAS operator version	8.0	

Online documentation and link summary

Complete documentation for IBM InfoSphere Information Server is available only through the product installation. Additional and more current documentation is available online. Table A-14 summarizes the online references that are required for the pre-installation tasks. These references are listed throughout this paper.

Table A-14 Online references for pre-installation tasks

Description	Web page
IBM Passport Advantage software downloads	http://www.ibm.com/software/howtobuy/passportadvantage/index.html
InfoSphere Information Server Information Center	http://publib.boulder.ibm.com/infocenter/iisinfsv/v8r0/index.jsp
InfoSphere Information Server release notes	http://publib.boulder.ibm.com/infocenter/iisinfsv/v8r0/topic/com.ibm.swg.im.iis.productization.iisinfsv.nav.doc/containers/cont_iisinfsrv_rnote.html
InfoSphere Information Server system requirements	http://www.ibm.com/support/docview.wss?uid=swg27008923
WebSphere Application Server 6.0 Info Center	http://publib.boulder.ibm.com/infocenter/wasinfo/v6r0/topic/com.ibm.websphere.base.doc/info/welcome_base.htm
WebSphere Application Server 6.0.2 release notes	http://publib.boulder.ibm.com/infocenter/wasinfo/v6r0/topic/com.ibm.websphere.base.doc/info/aes/ae/v6rn.html
WebSphere Application Server 6.0.2 hardware requirements	http://www.ibm.com/support/docview.wss?rs=180&uid=swg27007250
WebSphere Application Server 6.0.2 software requirements	http://www.ibm.com/support/docview.wss?rs=180&uid=swg27007256
DB2 v9.1 Information Center	http://publib.boulder.ibm.com/infocenter/db2luw/v9/index.jsp
DB2 v9.1 release notes	http://publib.boulder.ibm.com/infocenter/db2luw/v9/topic/com.ibm.db2.udb.doc/doc/c0023859.htm
DB2 v9 system requirements	http://publib.boulder.ibm.com/infocenter/db2luw/v9/topic/com.ibm.db2.udb.uprun.doc/doc/r0025127.htm

For IBM employees, the latest InfoSphere Information Server platform support and InfoSphere Information Server connectivity matrix are on the Xtreme Leverage Sales Portal at:

http://w3-103.ibm.com/software/xl/portal/viewcontent?type=doc&srcID=DM&docID=U585697X60278U90

Network ports used by InfoSphere Information Server

Table A-15 summarizes the list of network ports used by InfoSphere Information Server.

Table A-15 Network ports used by InfoSphere Information Server

Components	Description	Default port numbers
Metadata database - DB2 listener port	When using DB2 for xmeta repository	50000
Metadata Database - Oracle listener port	When using Oracle for xmeta repository	1521
InfoSphere Information Server agents	InfoSphere Information Server administration	31531, 31533
InfoSphere Information Server admin console	InfoSphere Information Server administration	9080 *
DataStage RPC daemon	DataStage client listener	31538 *
DataStage job monitor	Job Monitor listener port 1	3500
	Job Monitor listener port 2	13501
InfoSphere Information Server parallel engine	Engine conductor/section leader ($APT_PM_START_PORT)	>= 10000
	Engine player-to-player data transfer for cluster/grid ($APT_PLAYER_CONNECTION_PORT)	>= 11000

Components	Description	Default port numbers
Domain server ports (WebSphere Application Server)	Administrative console port (WC_adminhost)	9060
	HTTPS transport port (WC_defaulthost_secure)	9443
	Administrative Console secure port (WC_adminhost_secure)	9043
	Bootstrap port (BOOTSTRAP_ADDRESS)	2809 *
	Bootstrap port for deployment manager (BOOTSTRAP_ADDRESS)	9809
	SOAP connector port (SOAP_CONNECTOR_ADDRESS)	8880
	ORB listener port (ORB_LISTENER_ADDRESS)	9100 *
	SSL listener ports	9401-9403 *
	High availability manager communication port (DCS_UNICAST_ADDRESS)	9353
	Service integration port (SIB_ENDPOINT_ADDRESS)	7276
	Service integration secure port (SIB_ENDPOINT_SECURE_ADDRESS)	7286
	MQ transport port (SIB_MQ_ENDPOINT_ADDRESS)	5558
	MQ transport secure port (SIB_MQ_ENDPOINT_SECURE_ADDRESS)	5578

Glossary of terminology and abbreviations

Table A-16 provides terminology and abbreviations.

Table A-16 Terminology and abbreviations

Term	Definition
Domain	Application server (WebSphere Application Server) and deployed InfoSphere Information Server services
Engine	DataStage runtime engine (server and parallel)
Layer	A self-contained component of InfoSphere Information Server, for example, engine, domain, metadata repository, client
Metadata repository	Database used to store InfoSphere Information Server ("xmeta") design, configuration, and runtime metadata
Metadata server	Metadata repository and domain
Tier	A physical hardware server that might have one or more InfoSphere Information Server layers installed

Example user setup for UNIX environments

Table A-17 provides the UNIX commands that you can use to create the groups and users that are required to install and configure IBM InfoSphere Information Server.

Table A-17 User names and groups for specific user accounts

User account	Default user name	Primary group	Secondary group	Notes
DataStage administrator	dsadm	dstage		
DB2 administration server	dasusr1	dasadm1		Only needed for DB2.
DB2 instance owner	db2inst1	db2iadm1	dasadm1	Only needed for DB2.
DB2 fenced user	db2fenc1	db2fadm1		Only needed for DB2.
Metadata repository owner	xmeta	xmeta		DB2 uses OS authentication.
Information Analyzer analysis database owner	iauser	iauser		DB2 uses OS authentication.

To install and configure the listed groups and users, use the following commands:

- Groups

```
groupadd  db2iadm1
groupadd db2fadm1
groupadd dasadm1
groupadd dstage
```

- Users

```
useradd -gdb2fadm1 -pdb2fenc1 -m -d /home/db2fenc1 db2fenc1
useradd -gdb2iadm1 -Gdasadm1 -pdb2inst1 -m -d /home/db2inst1
db2inst1
useradd -gdasadm1 -pdasusr1 -m -d /home/dasusr1 dasusr1
useradd -gdstage -pdsadm -m -d /home/dsadm dsadm
useradd -piasuer -m -d /home/iauser iauser
useradd -pxmeta100 -m -d /home/xmeta xmeta
```

IBM InfoSphere Information Server Installation and Configuration Guide

Pre-installation checklists for a fast start implementation

Guidelines for planning and configuring your installation

Detailed product and platform information

This IBM Redpaper publication provides suggestions, hints and tips, directions, installation steps, checklists of prerequisites, and configuration information collected from several IBM InfoSphere Information Server experts. It is intended to minimize the time required to successfully install and configure the InfoSphere Information Server.

The information in this document is based on field experiences of experts who have implemented InfoSphere Information Server. As such, it is intended to supplement, and not replace, the product documentation.

Discover the proven choices and combinations for installing InfoSphere Information Server that have been the most successful for the IBM InfoSphere Center Of Excellence. This paper includes a broad range of customer needs and experiences, with a focus on the following areas:

- Information Server architecture
- Checklists
- Prerequisites
- Configuration choices that work well together

This paper is based on thousands of hours of production systems experience from which you can now reap significant benefits.

REDP-4596-00

INTERNATIONAL TECHNICAL SUPPORT ORGANIZATION

BUILDING TECHNICAL INFORMATION BASED ON PRACTICAL EXPERIENCE

IBM Redbooks are developed by the IBM International Technical Support Organization. Experts from IBM, Customers and Partners from around the world create timely technical information based on realistic scenarios. Specific recommendations are provided to help you implement IT solutions more effectively in your environment.

For more information:
ibm.com/redbooks