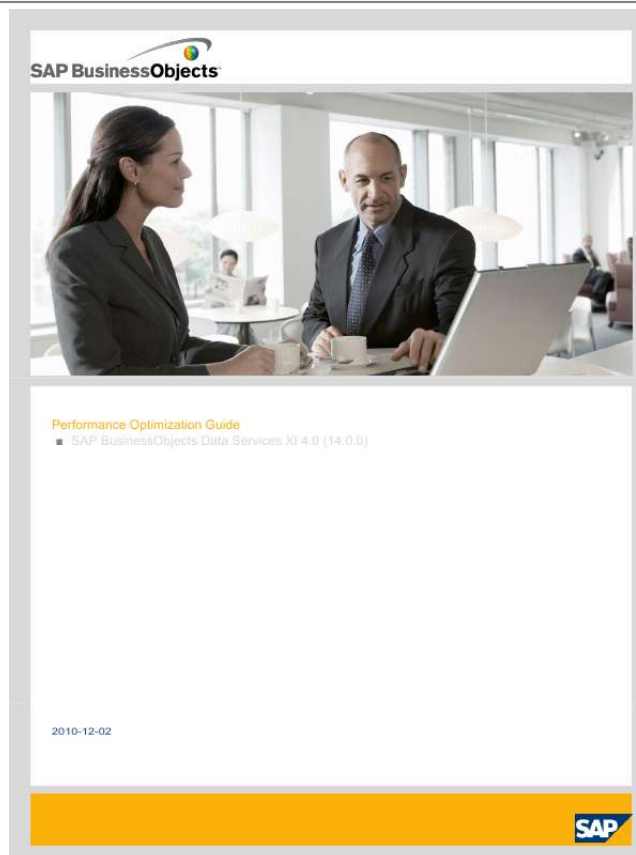




Your PDF Guides

You can read the recommendations in the user guide, the technical guide or the installation guide for BUSINESS OBJECTS DATA SERVICES XI 4.0. You'll find the answers to all your questions on the BUSINESS OBJECTS DATA SERVICES XI 4.0 in the user manual (information, specifications, safety advice, size, accessories, etc.). Detailed instructions for use are in the User's Guide.

User manual BUSINESS OBJECTS DATA SERVICES XI 4.0
User guide BUSINESS OBJECTS DATA SERVICES XI 4.0
Operating instructions BUSINESS OBJECTS DATA SERVICES XI 4.0
Instructions for use BUSINESS OBJECTS DATA SERVICES XI 4.0
Instruction manual BUSINESS OBJECTS DATA SERVICES XI 4.0



[You're reading an excerpt. Click here to read official BUSINESS OBJECTS DATA SERVICES XI 4.0 user guide](http://yourpdfguides.com/dref/3594853)
<http://yourpdfguides.com/dref/3594853>

Manual abstract:

@@Data contained in this document serves informational purposes only. National product specifications may vary. These materials are subject to change without notice. These materials are provided by SAP AG and its affiliated companies ("SAP Group") for informational purposes only, without representation or warranty of any kind, and SAP Group shall not be liable for errors or omissions with respect to the materials. The only warranties for SAP Group products and services are those that are set forth in the express warranty statements accompanying such products and services, if any. Nothing herein should be construed as constituting an additional warranty. 2010-12-02 Contents Chapter 1 Welcome to SAP BusinessObjects Data Services...

.....
.....
.....

.....
.....
.....
.....

.....
..7 Welcome...

.....
.....

.....
.....
.....
.....

.....
.....
.....
.....

.....
.....
.....
.....

.....
.....
.....

..7 Documentation set for SAP BusinessObjects Data Services.....

.....
.....
.....
.....

.....
..7 Accessing documentation...

.....
.....

.....
.....
.....
.....

.....
.....
.....
.....

.....
.....
.....

....10 Accessing documentation on Windows.....

.....

.....
.....
.....

.....
.....
.....
.....

.....
.....

..10 Accessing documentation on UNIX....

.....

.....
.....
.....

.....
.....
.....
.....

.....

....10 Accessing documentation from the Web.

.....
.....

.....
.....
.....

.....
.....
.....
.....

....10 SAP BusinessObjects information resources.....

.....
.....
.....

.....
.....
.....
.....

.....
.....

..11 Environment Test Strategy...

.....

.....
.....
.....

.....
.....
.....
.....

.....
.....
.....

.....13 The source OS and database server...

.....
.....
.....

.....
.....
.....

.....
.....
.....

...13 Operating system.....

.....
.....
.....

.....
.....
.....

.....
.....
.....

.....
.....

...13 Database..

.....
.....
.....
.....

.....
.....
.....

.....
.....
.....

.....
.....
.....

.....14 The target OS and database server.....

.....
.....
.....

.....
.....
.....

.....
.....
.....
...14 Operating system.....

.....
.....
.....

.....
.....
.....

.....
.....
.....

.....
.....

...14 Database..

.....
.....
.....
.....

.....
.....
.....

.....
.....
.....

.....
.....
.....

.....14 The network.....

.....
.....
.....

.....
.....
.....

.....
.....
.....

.....
.....
.....

.....15 Job Server OS and job options..

.....
.....

.....
.....
.....
.....
.....
.....

..15 Operating system.....

.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....

.....15 Jobs...

.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....

..16 Measuring Performance...

.....
.....
.....
.....
.....
.....
.....
.....

.....
.....
.....
.....
.....19 Data Services processes and threads....

.....
.....
.....

.....
.....
.....

.....
.....
.....

19 Processes

.....
.....
.....

.....
.....
.....

.....
.....
.....

.....
.....
.....

.....19 Threads....

.....
.....
.....

.....
.....
.....

.....
.....
.....

.....
.....
.....

.....20 Measuring performance of jobs....

.....
.....
.....
.....

.....
.....
.....
.....
.....
.....
.....

.....20 *Checking system utilization.*

.....
.....
.....
.....
.....
.....
.....
.....
.....

....20 *Analyzing log files for task duration.*

.....
.....
.....
.....
.....
.....
.....

..24 *Reading the Monitor Log for execution statistics ...*

.....
.....
.....
.....
.....
.....
.....

....25 *Reading the Performance Monitor for execution statistics.....*

.....
.....
.....
.....
.....

...26 *To view the Performance Monitor..*

.....
.....
.....

.....

.....

.....

.....

.....

.....

.....

...26 Reading Operational Dashboards for execution statistics..

.....

.....

.....

.....

.....

.....

.....

.....

...27 1.1 1.

2 1.3 1.3.1 1.3.

2 1.3.3 1.4 Chapter 2 2.1 2.1.1 2.1.2 2.2 2.

2.1 2.2.2 2.3 2.4 2.4.1 2.4.2 Chapter 3 3.

1 3.1.1 3.1.2 3.

2 3.2.1 3.2.2 3.

2.3 3.2.4 3.2.5 3.2.6 3 2010-12-02 Contents 3.2.7 Chapter 4 To compare execution times for the same job over time.

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....27 Tuning Overview....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.29 Strategies to execute jobs.....

.....

.....

.....

.....

.....

.....

.....
.....
.....
.....

..29 Maximizing push-down operations to the database server.....

.....
.....
.....

.....
.....
.....

.....29 Improving throughput...

.....
.....
.....
.....

.....
.....
.....
.....

.....
.....
.....
.....

.....
.....

30 Using advanced tuning options.....

.....
.....
.....
.....
.....

.....
.....
.....
.....

.....
.....
.....

30 Maximizing Push-Down Operations.....

.....
.....
.....
.....

.....
.....
.....
.....

.....
.....

.33 Push-down operations....

.....
.....
.....
.....

.....
.....
.....
.....
.....
.....
.....
.....
.....
.....

....33 Full push-down operations.....

.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....

.33 Partial push-down operations....

.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....

...34 Operations that cannot be pushed down.....

.....
.....
.....
.....
.....
.....
.....
.....
.....
.....

...34 Push-down examples..

.....
.....
.....
.....
.....
.....

.....
.....
.....
.....
.....
.....
.....
.....

..35 Collapsing transforms to push down operations example.....

.....
.....
.....

.....
.....
.....

.....35 Full push down from the source to the target example...

.....
.....
.....
.....

.....
.....
.....

.....36 Full push down for auto correct load to the target example

.....
.....
.....
.....

.....
.....
.....

..36 Partial push down to the source example...

.....
.....
.....
.....
.....

.....
.....
.....
.....

.....37 Push-down of SQL join example

.....
.....
.....

.....
.....
.....
.....

.....
.....
.....
.....

.....37 To view SQL..

.....
.....
.....

.....
.....
.....

.....
.....
.....

.....
.....
.....

.....
38 Data_Transfer transform for push-down operations.....

.....
.....

.....
.....
.....

.....
....39 Push down an operation after a blocking operation example.

.....
.....

.....
.....
.....

.....
.....39 Using Data_Transfer tables to speed up auto correct loads example.....

.....
.....
.....

.....
...42 Database link support for push-down operations across datastores..

.....
.....

.....
.....
.....

..43 Software support.....

.....
.....
.....

.....
.....
.....

.....
.....
.....

.....
.....
.....
.44 To take advantage of linked datastores...
.....

.....
.....
.....
.....
.....
.....44 Example of push-down with linked datastores.
.....

.....
.....
.....
.....
.....
.....
.....
.45 Generated SQL statements.....
.....

.....
.....
.....
.....
.....
.....
.....
46 Tuning performance at the data flow or Job Server level.....
.....

.....
.....
.....
.....
.....
.....
.....
.....46 Using Caches....
.....

.....
.....
.....
.....
.....
.....
.....
.....
.....

.....
.....
.....
.....

.....
.....
.....
.....
.....49 Caching data....

.....
.....
.....
.....

.....
.....
.....
.....

.....
.....
.....
.....

.....
.....
.....

.....49 Caching sources..

.....
.....
.....
.....

.....
.....
.....
.....

.....
.....
.....
.....

.....
.....
.....

....49 Caching joins.....

.....
.....
.....
.....

.....
.....
.....
.....

.....
.....
.....
.....

.....
.....

.....
.....50 Changing cache type for a data flow

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.51 Caching lookups.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

..52 Caching table comparisons...

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....53 Specifying a pageable cache directory...

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

53 Using persistent cache.....

.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....

.....54 Using persistent cache tables as sources ...

.....
.....
.....
.....
.....
.....
.....
.....
.....
.....

....55 4.1 4.1.1 4.1.2 4.

1.3 Chapter 5 5.1 5.1.1 5.1.2 5.1.3 5.2 5.
2.1 5.2.2 5.2.
3 5.2.4 5.2.5 5.
3 5.4 5.4.1 5.4.2 5.5 5.5.1 5.5.

2 5.5.3 5.5.4 5.5.5 Chapter 6 6.1 6.1.1 6.
1.2 6.1.3 6.1.
4 6.1.5 6.1.6 6.

2 6.2.1 4 2010-12-02 Contents 6.3 6.3.1 6.3.2 6.3.3 Chapter 7 Monitoring and tuning caches.

.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....

.....55 Using statistics for cache self-tuning

.....
.....
.....
.....
.....
.....
.....

.....
.....
.....

..55 To automatically choose the cache type.....

.....
.....
.....

.....
.....
.....

.....
.....
.....

..55 To monitor and tune in-memory and pageable caches.....

.....
.....
.....

.....
.....
.....

..56 Using Parallel Execution.....

.....
.....
.....

.....
.....
.....

.....
.....
.....

.....59 Parallel data flows and work flows.....

.....
.....
.....

.....
.....
.....

.....
.....
.....

.....59 Parallel execution in data flows.....

.....
.....
.....

.....
.....
.....

.....
.....
.....
.....

60 Table partitioning.....

.....
.....
.....
.....

.....
.....
.....

.....
.....
.....

.....60 Degree of parallelism

.....
.....

.....
.....
.....

.....
.....
.....

.....
.....

....64 Combining table partitioning and a degree of parallelism.....

.....
.....
.....

.....
.....

70 File multi-threading.....

.....
.....
.....

.....
.....
.....

.....
.....
.....
.....

.....
...72 Distributing Data Flow Execution..

.....
.....
.....
.....
.....
.....
.....
.....
.....
.....

.....
...77 Splitting a data flow into sub data flows..

.....
.....
.....
.....
.....
.....
.....
.....
.....

.....
....77 Run as a separate process option

.....
.....
.....
.....
.....
.....
.....
.....
.....
.....

.....
....77 Examples of multiple processes for a data flow.....

.....
.....
.....
.....
.....
.....
.....

.....
....78 Data_Transfer transform.

.....
.....
.....
.....
.....

.....
.....

.....
.....
.....

.....
.....
.....

.....83 Examples of multiple processes with Data_Transfer..

.....
.....
.....

.....
.....
.....

.83 Using grid computing to distribute data flow execution.....

.....
.....
.....

.....
.....
.....

86 Server Group.....

.....
.....
.....

.....
.....
.....

.....
.....
.....

.....
.....
.....

.87 Distribution levels for data flow execution.....

.....
.....
.....

.....
.....
.....

.....
.....

....87 Using Bulk Loadi.

.....

.....
..107 Configuring bulk loading for Netezza ...

.....
.....

.....
.....
.....

.....
.....
.....

.....107 To configure the target table.....

.....
.....
.....

.....
.....
.....

.....
.....
.....

.....
..108 Bulk loading in Sybase ASE....

.....
.....

.....
.....
.....

.....
.....
.....

.....
.....

...108 Bulk loading in Sybase IQ..

.....
.....

.....
.....
.....

.....
.....
.....

.....
.....
.....

.....109 Bulk loading in Teradata.

.....

.....
.....
.....

.....
.....
.....
.....

.....
.....
.....
.....

.....
.....
.....
.....

..109 When to use each Teradata bulk load method...

.....
.....
.....
.....

.....
.....
.....

..110 How SAP BusinessObjects Data Services and Teradata use the file options to load.....

....114 Using the UPSERT bulk operation.....

.....
.....
.....

.....
.....
.....
.....

.....
.....
.....
.....

..116 Parallel Transporter method.....

.....
.....
.....

.....
.....
.....
.....

.....
.....
.....
.....

.....117 Teradata standalone utilities .

.....
.....
.....

.....
.....
.....

.....
.....
.....
.....

120 Other Tuning Techniques.....

.....
.....
.....

.....
.....
.....

.....
.....
.....

.....131 Source-based performance options.....

.....
.....
.....

.....
.....
.....

.....
.....
.....

.131 Join ordering.....

.....
.....
.....

.....
.....
.....

.....
.....
.....

.....
.....
.....

.132 Minimizing extracted data.....

.....
.....
.....

.....
.....
.....
.....
.....
.....
.....
.....

..138 Using array fetch size.....

.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....

....138 Target-based performance options.

.....
.....
.....
.....
.....
.....
.....
.....
.....

.139 Loading method

.....
.....
.....
.....
.....
.....
.....
.....
.....
.....

140 Rows per commit.....

.....
.....
.....
.....

.....
.....
.....
.....
.....
.....
.....
.....
.....
.....

...141 Job design performance options.....

.....
.....
.....
.....
.....
.....
.....
.....
.....
.....

...141 Loading only changed data

.....
.....
.....
.....
.....
.....
.....
.....
.....
.....

.141 Minimizing data type conversion....

.....
.....
.....
.....
.....
.....
.....
.....
.....
.....

..142 Minimizing locale conversion...

.....
.....
.....
.....
.....
.....

Information about how to improve the performance of SAP BusinessObjects Data Services. Detailed reference material for SAP BusinessObjects Data Services Designer. Important information you need before installing and deploying this version of SAP BusinessObjects Data Services.

A compiled "master" PDF of core SAP BusinessObjects Data Services books containing a searchable master table of contents and index: · Administrator's Guide · Designer Guide · Reference Guide · Management Console Guide · Performance Optimization Guide · Supplement for J.D. Edwards · Supplement for Oracle Applications · Supplement for PeopleSoft · Supplement for Salesforce.com · Supplement for Siebel · Supplement for SAP Information about building dictionaries and extraction rules to create your own extraction patterns to use with Text Data Processing transforms. Information about the linguistic analysis and extraction processing features that the Text Data Processing component provides, as well as a reference section for each language supported.

A step-by-step introduction to using SAP BusinessObjects Data Services. Management Console Guide Performance Optimization Guide Reference Guide Release Notes Technical Manuals Text Data Processing Extraction Customization Guide Text Data Processing Language Reference Guide Tutorial 8 2010-12-02 Welcome to SAP BusinessObjects Data Services Document What this document provides Upgrade Guide Release-specific product behavior changes from earlier versions of SAP BusinessObjects Data Services to the latest release. This manual also contains information about how to migrate from SAP BusinessObjects Data Quality Management to SAP BusinessObjects Data Services. Highlights of new key features in this SAP BusinessObjects Data Services release. This document is not updated for support package or patch releases.

What's New In addition, you may need to refer to several Adapter Guides and Supplemental Guides. Document What this document provides Supplement for J.D. Edwards Supplement for Oracle Applications Supplement for PeopleSoft Supplement for Salesforce.com Supplement for SAP Supplement for Siebel Information about interfaces between SAP BusinessObjects Data Services and J.D. Edwards World and J.D. Edwards OneWorld. Information about the interface between SAP BusinessObjects Data Services and Oracle Applications.

Information about interfaces between SAP BusinessObjects Data Services and PeopleSoft. Information about how to install, configure, and use the SAP BusinessObjects Data Services Salesforce.com Adapter Interface. Information about interfaces between SAP BusinessObjects Data Services, SAP Applications, and SAP NetWeaver BW. Information about the interface between SAP BusinessObjects Data Services and Siebel. We also include these manuals for information about SAP BusinessObjects Information platform services. Document What this document provides Information platform services Administrator's Guide Information platform services Installation Guide for UNIX Information platform services Installation Guide for Windows Information for administrators who are responsible for configuring, managing, and maintaining an Information platform services installation. Installation procedures for SAP BusinessObjects Information platform services on a UNIX environment. Installation procedures for SAP BusinessObjects Information platform services on a Windows environment. 9 2010-12-02 Welcome to SAP BusinessObjects Data Services 1.

3 Accessing documentation You can access the complete documentation set for SAP BusinessObjects Data Services in several places. 1.3.1 Accessing documentation on Windows After you install SAP BusinessObjects Data Services, you can access the documentation from the Start menu. 1.

Choose Start > Programs > SAP BusinessObjects Data Services XI 4.0 > Data Services Documentation. Note: Only a subset of the documentation is available from the Start menu. The documentation set for this release is available in <LINK_DIR>\DocBooks\en. 2.

Click the appropriate shortcut for the document that you want to view. 1.3.2 Accessing documentation on UNIX After you install SAP BusinessObjects Data Services, you can access the online documentation by going to the directory where the printable PDF files were installed. 1. Go to <LINK_DIR>/doc/book/en/. 2. Using Adobe Reader, open the PDF file of the document that you want to view. 1.3.

3 Accessing documentation from the Web You can access the complete documentation set for SAP BusinessObjects Data Services from the SAP BusinessObjects Business Users Support site. 1. Go to <http://help.sap.com>. 2. Click SAP BusinessObjects at the top of the page. 3. Click All Products in the navigation pane on the left. 10 2010-12-02 Welcome to SAP BusinessObjects Data Services You can view the PDFs online or save them to your computer.

1.4 SAP BusinessObjects information resources A global network of SAP BusinessObjects technology experts provides customer support, education, and consulting to ensure maximum information management benefit to your business. Useful addresses at a glance: 11 2010-12-02 Welcome to SAP BusinessObjects Data Services Address Content Customer Support, Consulting, and Education services <http://service.sap.com/> Information about SAP Business User Support programs, as well as links to technical articles, downloads, and online forums.

Consulting services can provide you with information about how SAP BusinessObjects can help maximize your information management investment. Education services can provide information about training options and modules. From traditional classroom learning to targeted e-learning seminars, SAP BusinessObjects can offer a training package to suit your learning needs and preferred learning style. SAP BusinessObjects Data Services Community Get online and timely information about SAP BusinessObjects Data Services, including tips and tricks, <http://www.sdn.sap.com/irj/sdn/ds>

additional downloads, samples, and much more. All content is to and from the community, so feel free to join in and contact us if you have a submission. Forums on SCN (SAP Community Network) <http://forums.sdn.sap.com/forum.jspa?forumID=305> Search the SAP BusinessObjects forums on the SAP Community Network to learn from other SAP BusinessObjects Data Services users and start posting questions or share your knowledge with the community. Blueprints for you to download and modify to fit your needs. Each blueprint contains the necessary SAP BusinessObjects Data Services project, jobs, data flows, file formats, sample data, template tables, and custom functions to run the data flows in your environment with only a few modifications.

SAP BusinessObjects product documentation. Blueprints <http://www.sdn.sap.com/irj/boc/blueprints> Product documentation <http://help.sap.com/>



[You're reading an excerpt. Click here to read official BUSINESS OBJECTS DATA SERVICES XI 4.0 user guide](http://yourpdfguides.com/dref/3594853)
<http://yourpdfguides.com/dref/3594853>

com/businessobjects/ Supported Platforms (Product Availability Matrix) <https://service.sap.com/PAM> Get information about supported platforms for SAP BusinessObjects Data Services. Use the search function to search for Data Services.

Click the link for the version of Data Services you are searching for. 12 2010-12-02 Environment Test Strategy Environment Test Strategy This section covers suggested methods of tuning source and target database applications, their operating systems, and the network used by your SAP BusinessObjects Data Services environment. It also introduces key job execution options. This section contains the following topics: . . . The source OS and database server The target OS and database server The network Job Server OS and job options To test and tune jobs, work with all four of these components in the order shown above. In addition to the information in this section, you can use your UNIX or Windows operating system and database server documentation for specific techniques, commands, and utilities that can help you measure and tune the SAP BusinessObjects Data Services environment.

2.1 The source OS and database server Tune the source operating system and database to quickly read data from disks. 2.1.1 Operating system Make the input and output (I/O) operations as fast as possible.

The read-ahead protocol, offered by most operating systems, can greatly improve performance. This protocol allows you to set the size of each I/O operation.

Usually its default value is 4 to 8 kilobytes which is too small. Set it to at least 64K on most platforms. 13 2010-12-02 Environment Test Strategy 2.1.2 Database Tune your database on the source side to perform SELECTs as quickly as possible. In the database layer, you can improve the performance of SELECTs in several ways, such as the following: . . . Create indexes on appropriate columns, based on your data flows. Increase the size of each I/O from the database server to match the OS read-ahead I/O size. Increase the size of the shared buffer to allow more data to be cached in the database server.

Cache tables that are small enough to fit in the shared buffer. For example, if jobs access the same piece of data on a database server, then cache that data. Caching data on database servers will reduce the number of I/O operations and speed up access to database tables. See your database server documentation for more information about techniques, commands, and utilities that can help you measure and tune the the source databases in your jobs. 2.2 The target OS and database server Tune the target operating system and database to quickly write data to disks. 2.2.1 Operating system Make the input and output operations as fast as possible. For example, the asynchronous I/O, offered by most operating systems, can greatly improve performance.

Turn on the asynchronous I/O. 2.2.2 Database Tune your database on the target side to perform INSERTs and UPDATES as quickly as possible. In the database layer, there are several ways to improve the performance of these operations.

Here are some examples from Oracle: 14 2010-12-02 Environment Test Strategy . . . Turn off archive logging Turn off redo logging for all tables Tune rollback segments for better performance Place redo log files and data files on a raw device if possible Increase the size of the shared buffer See your database server documentation for more information about techniques, commands, and utilities that can help you measure and tune the the target databases in your jobs. 2.3 The network When reading and writing data involves going through your network, its ability to efficiently move large amounts of data with minimal overhead is very important. Do not underestimate the importance of network tuning (even if you have a very fast network with lots of bandwidth). Set network buffers to reduce the number of round trips to the database servers across the network.

For example, adjust the size of the network buffer in the database client so that each client request completely fills a small number of network packets. 2.4 Job

Server OS and job options Tune the Job Server operating system and set job execution options to improve performance and take advantage of self-tuning features of SAP BusinessObjects Data Services. 2.4.1 Operating system SAP BusinessObjects Data Services jobs are multi-threaded applications. Typically a single data flow in a job initiates one al_engine process that in turn initiates at least 4 threads. For maximum performance benefits: . . Consider a design that will run one al_engine process per CPU at a time. Tune the Job Server OS so that threads spread to all available CPUs. For more information, see Checking system utilization.

15 2010-12-02 Environment Test Strategy 2.4.2 Jobs You can tune job execution options after: . . . Tuning the database and operating system on the source and the target computers Adjusting the size of the network buffer Your data flow design seems optimal You can tune the following execution options to improve the performance of your jobs: . . Monitor sample rate Collect statistics for optimization and Use collected statistics 2.4.2.1 Setting Monitor sample rate During job execution, the SAP BusinessObjects Data Services writes information to the monitor log file and updates job events after processing the number of rows specified in Monitor sample rate. Default value is 1000. Increase Monitor sample rate to reduce the number of calls to the operating system to write to the log file. When setting Monitor sample rate, you must evaluate performance improvements gained by making fewer calls to the operating system against your ability to view more detailed statistics during job execution. With a higher Monitor sample rate, the software collects more data before calling the operating system to open the file, and performance improves.

However, with a higher monitor rate, more time passes before you can view statistics during job execution. In production environments when your jobs transfer large volumes of data, you should increase Monitor sample rate to 50,000. Note: If you use a virus scanner on your files, exclude the SAP BusinessObjects Data Services log from the virus scan. Otherwise, the virus scan analyzes the log repeatedly during the job execution, which causes a performance degradation. 2.

4.2.2 Collecting statistics for self-tuning 16 2010-12-02 Environment Test Strategy SAP BusinessObjects Data Services provides a self-tuning feature to determine the optimal cache type (in-memory or pageable) to use for a data flow. 2.4.

2.3 To take advantage of this self-tuning feature 1. When you first execute a job, select the option Collect statistics for optimization to collect statistics which include number of rows and width of each row.



[You're reading an excerpt. Click here to read official BUSINESS OBJECTS DATA SERVICES XI 4.0 user guide](http://yourpdfguides.com/dref/3594853)
<http://yourpdfguides.com/dref/3594853>

Ensure that you collect statistics with data volumes that represent your production environment. This option is not selected by default. 2. The next time you execute the job, this option is selected by default. 3. When changes occur in data volumes, re-run your job with Collect statistics for optimization to ensure that the software has the most current statistics to optimize cache types. For more information about these caches, see .

Related Topics · Using Caches 17 2010-12-02 Environment Test Strategy 18 2010-12-02 Measuring Performance Measuring Performance This section contains the following topics: · · Data Services processes and threads Measuring performance of jobs 3.1 Data Services processes and threads Data Services uses processes and threads to execute jobs that extract data from sources, transform the data, and load data into a data warehouse. The number of concurrently executing processes and threads affects the performance of Data Services jobs. 3.1.1 Processes The processes used to run jobs are: · al_jobserver The al_jobserver initiates one process for each Job Server configured on a computer. This process does not use much CPU power because it is only responsible for launching each job and monitoring the job's execution. · al_engine For batch jobs, an al_engine process runs when a job starts and for each of its data flows. Real-time jobs run as a single process. The number of processes a batch job initiates also depends upon the number of: · · parallel work flows parallel data flows sub data flows For an example of the monitor log that displays the processes, see Analyzing log files for task duration. 19 2010-12-02 Measuring Performance 3.1.2 Threads A data flow typically initiates one al_engine process, which creates one thread per data flow object. A data flow object can be a source, transform, or target. For example, two sources, a query, and a target could initiate four threads.

If you are using parallel objects in data flows, the thread count will increase to approximately one thread for each source or target table partition. If you set the Degree of parallelism (DOP) option for your data flow to a value greater than one, the thread count per transform will increase. For example, a DOP of 5 allows five concurrent threads for a Query transform. To run objects within data flows in parallel, use the following features: · · Table partitioning File multithreading Degree of parallelism for data flows Related Topics · Using Parallel Execution 3.2 Measuring performance of jobs You can use several techniques to measure performance of SAP BusinessObjects Data Services jobs: · · · · Checking system utilization Analyzing log files for task duration Reading the Monitor Log for execution statistics Reading the Performance Monitor for execution statistics Reading Operational Dashboards for execution statistics 3.

2.1 Checking system utilization The number of processes and threads concurrently executing affects the utilization of system resources (see Data Services processes and threads). 20 2010-12-02 Measuring Performance Check the utilization of the following system resources: · · · · CPU Memory Disk Network To monitor these system resources, use the following tools: For UNIX: · top or a third party utility (such as glance for HP-UX) For Windows: · Performance tab on the Task Manager Depending on the performance of your jobs and the utilization of system resources, you might want to adjust the number of processes and threads. The following sections describe different situations and suggests features to adjust the number of processes and threads for each situation. 3.2.1.1 CPU utilization SAP BusinessObjects Data Services is designed to maximize the use of CPUs and memory available to run the job. The total number of concurrent threads a job can run depends upon job design and environment. Test your job while watching multi-threaded processes to see how much CPU and memory the job requires.

Make needed adjustments to your job design and environment and test again to confirm improvements. For example, if you run a job and see that the CPU utilization is very high, you might decrease the DOP value or run less parallel jobs or data flows. Otherwise, CPU thrashing might occur. For another example, if you run a job and see that only half a CPU is being used, or if you run eight jobs on an eight-way computer and CPU usage is only 50%, you can be interpret this CPU utilization in several ways: · · One interpretation might be that the software is able to push most of the processing down to source and/or target databases. Another interpretation might be that there are bottlenecks in the database server or the network connection. Bottlenecks on database servers do not allow readers or loaders in jobs to use Job Server CPUs efficiently. To determine bottlenecks, examine: · Disk service time on database server computers 21 2010-12-02 Measuring Performance Disk service time typically should be below 15 milliseconds. Consult your server documentation for methods of improving performance. For example, having a fast disk controller, moving database server log files to a raw device, and increasing log size could improve disk service time. · Number of threads per process allowed on each database server operating system.

For example: · On HP-UX, the number of kernel threads per process is configurable. The CPU to thread ratio defaults to one-to-one. It is recommended that you set the number of kernel threads per CPU to between 512 and 1024. On Solaris and AIX, the number of threads per process is not configurable. The number of threads per process depends on system resources.

If a process terminates with a message like "Cannot create threads," you should consider tuning the job. For example, use the Run as a separate process option to split a data flow or use the Data Transfer transform to create two sub data flows to execute sequentially. Since each sub data flow is executed by a different al_engine process, the number of threads needed for each will be 50% less than in your previous job design. If you are using the Degree of parallelism option in your data flow, reduce the number for this option in the data flow Properties window. · Network connection speed Determine the rate that your data is being transferred across your network.

· · If the network is a bottle neck, you might change your job execution distribution level from sub data flow to data flow or job to execute the entire data flow on the local Job Server. If the capacity of your network is much larger, you might retrieve multiple rows from source databases using fewer requests. · · Yet another interpretation might be that the system is under-utilized.



[You're reading an excerpt. Click here to read official BUSINESS OBJECTS DATA SERVICES XI 4.0 user guide](http://yourpdfguides.com/dref/3594853)
<http://yourpdfguides.com/dref/3594853>

In this case, you might increase the value for the Degree of parallelism option and increase the number of parallel jobs and data flows. *Related Topics · Using Parallel Execution · Using grid computing to distribute data flow execution · Using array fetch size* 3.2.1.2 Memory For memory utilization, you might have one of the following different cases: · Low amount of physical memory. In this case, you might take one of the following actions: 22 2010-12-02 Measuring Performance · Add more memory to the Job Server. Redesign your data flow to run memory-consuming operations in separate sub data flows that each use a smaller amount of memory, and distribute the sub data flows over different Job Servers to access memory on multiple machines.

For more information, see *Splitting a data flow into sub data flows*. Redesign your data flow to push down memory-consuming operations to the database. For more information, see *Push-down operations*. · For example, if your data flow reads data from a table, joins it to a file, and then groups it to calculate an average, the group by operation might be occurring in memory. If you stage the data after the join and before the group by into a database on a different computer, then when a sub data flow reads the staged data and continues with the group processing, it can utilize memory from the database server on a different computer. This situation optimizes your system as a whole. For information about how to stage your data, see *Data_Transfer transform*. For more information about distributing sub data flows to different computers, see *Using grid computing to distribute data flow execution*. · Large amount of memory but it is under-utilized. In this case, you might cache more data.

Caching data can improve the performance of data transformations because it reduces the number of times the system must access the database. There are two types of caches available: in-memory and pageable. For more information, see *Caching data*. · Paging occurs. Pageable cache is the default cache type for data flows.

On Windows and Linux, the virtual memory available to the al_engine process is 1.5 gigabytes (500 megabytes of virtual memory is reserved for other engine operations, totaling 2GB). On UNIX, the virtual memory is limited for the al_engine process to 3.5 gigabytes (500MB is reserved for other engine operations, totaling 4GB). If more memory is needed than these virtual memory limits, the software starts paging to continue executing the data flow.

If your job or data flow requires more memory than these limits, you can take advantage of one of the following features to avoid paging: · Split the data flow into sub data flows that can each use the amount of memory set by the virtual memory limits. Each data flow or each memory-intensive operation within a data flow can run as a separate process that uses separate memory from each other to improve performance and throughput. For more information, see *Splitting a data flow into sub data flows*. · Push-down memory-intensive operations to the database server so that less memory is used on the Job Server computer. For more information, see *Push-down operations*. 23 2010-12-02 Measuring Performance 3.2.2 Analyzing log files for task duration The trace log shows the progress of an execution through each component (object) of a job. The following sample Trace log shows a separate Process ID (Pid) for the Job, data flow, and each of the two sub data flows. This sample log contains messages about sub data flows, caches, and statistics.

Related Topics · Splitting a data flow into sub data flows · Caching data · Reference Guide: Objects, Log 24 2010-12-02 Measuring Performance 3.2.3 Reading the Monitor Log for execution statistics The Monitor log file indicates how many rows SAP BusinessObjects Data Services produces or loads for a job. By viewing this log during job execution, you can observe the progress of row-counts to determine the location of bottlenecks. You can use the Monitor log to answer questions such as the following: · What transform is running at any moment? How many rows have been processed so far? The frequency that the Monitor log refreshes the statistics is based on Monitor sample rate. · How long does it take to build the cache for a lookup or comparison table? How long does it take to process the cache? If take long time to build the cache, use persistent cache. · How long does it take to sort? If take long time to sort, you can redesign your data flow to push down the sort operation to the database. · How much time elapses before a blocking operation sends out the first row? If your data flow contains resource-intensive operations after the blocking operation, you can add Data_Transfer transforms to push-down the resource-intensive operations. You can view the Monitor log from the following tools: · The Designer, as the job executes, when you click the Monitor icon. The Administrator of the Management Console, when you click the Monitor link for a job from the Batch Job Status page.

The Monitor log in the Designer shows the path for each object in the job, the number of rows processed, and the elapsed time for each object. The Absolute time column displays the total time from the start of the job to when the software completes the execution of the data flow object. *Related Topics · Setting Monitor sample rate · Using persistent cache · Push-down operations · Data_Transfer transform for push-down operations · Reference Guide: Objects, Log* 25 2010-12-02 Measuring Performance 3.2.4 Reading the Performance Monitor for execution statistics The Performance Monitor displays execution information for each work flow, data flow, and sub data flow within a job.

You can display the execution times in a table format. You can use the Performance Monitor to answer questions such as the following: · Which data flows might be bottlenecks? How much time did a data flow or sub data flow take to execute? How many rows did the data flow or sub data flow process? How much memory did a specific data flow use? Note: Memory statistics (Cache Size column) display in the Performance Monitor only if you select the Collect statistics for monitoring option when you execute the job. The following sample Performance Monitor shows the following information: · The Query_Lookup transform used 110 kilobytes of memory. The first sub data flow processed 830 rows, and the second sub data flow processed 35 rows. 3.

2.5 To view the Performance Monitor 1. Access the Management Console with one of the following methods: 26 2010-12-02 Measuring Performance · In the Designer top menu bar, click Tools and select Management Console. Click Start > Programs > SAP BusinessObjects Data Services XI 4.



[You're reading an excerpt. Click here to read official BUSINESS OBJECTS DATA SERVICES XI 4.0 user guide](http://yourpdfguides.com/dref/3594853)
<http://yourpdfguides.com/dref/3594853>

0 > Data Services Management Console. 2. On the launch page, click Administrator. 3. Select Batch > repository 4. On the Batch Job Status page, find a job execution instance.

5. Under Job Information for an instance, click Performance Monitor. Related Topics · To monitor and tune in-memory and pageable caches 3.2.6 Reading Operational Dashboards for execution statistics Operational dashboard reports contain job and data flow execution information for one or more repositories over a given time period (for example the last day or week). You can use operational statistics reports to answer some of the following questions: · · · · Are jobs executing within the allotted time frames? How many jobs succeeded or failed over a given execution period? How is the execution time for a job evolving over time? How many rows did the data flow process? 3.2.7 To compare execution times for the same job over time 1. Open the Management Console via one of the following methods: · · In the Designer top menu bar, choose Tools > Management Console. Choose Start > Programs > SAP BusinessObjects Data Services XI x.

x > Data Services Management Console. 2. On the launch page, click Operational Dashboard. 3. Look at the graphs in Job Execution Statistic History or Job Execution Duration History to see if performance is increasing or decreasing.

4. On the Job Execution Duration History page, if there is a specific day that looks high or low compared to the other execution times, click that point on the graph to view the Job Execution Duration graph for all of the jobs that ran that day. 5. Click View all history to compare different executions of a specific job or data flow. 6.

On the Job Execution History tab, you can select a specific job and number of days. 27 2010-12-02 Measuring Performance 7. On the Data Flow Execution History tab, you can select a specific job and number of days, as well as search for a specific data flow. Related Topics · Management Console Guide: Operational Dashboard Reports 28 2010-12-02 Tuning Overview Tuning Overview This section presents an overview of the different Data Services tuning options, with cross-references to subsequent chapters for more details. 4.1 Strategies to execute jobs 4.1.1 Maximizing push-down operations to the database server SAP BusinessObjects Data Services generates SQL SELECT statements to retrieve the data from source databases. The software automatically distributes the processing workload by pushing down as much as possible to the source database server. Pushing down operations provides the following advantages: · Use the power of the database server to execute SELECT operations (such as joins, Group By, and common functions such as decode and string functions).

Often the database is optimized for these operations. Minimize the amount of data sent over the network. Fewer rows can be retrieved when the SQL statements include filters or aggregations. · You can also do a full push down from the source to the target, which means the software sends SQL INSERT INTO... SELECT statements to the target database. The following features enable a full push down: · · Data_Transfer transform Database links and linked datastores Related Topics · Maximizing Push-Down Operations 29 2010-12-02 Tuning Overview 4.1.2 Improving throughput Use the following features to improve throughput: · Using caches for faster access to data You can improve the performance of data transformations by caching as much data as possible. By caching data in memory, you limit the number of times the system must access the database. · Bulk loading to the target The software supports database bulk loading engines including the Oracle bulk load API. You can have multiple bulk load processes running in parallel. · Other tuning techniques · Source-based performance options · · · · Join ordering Minimizing extracted data Using array fetch size Target-based performance options · · Loading method Rows per commit · Job design performance options · · · · Loading only changed data Minimizing data type conversion Minimizing locale conversion Precision in operations 4.1.

3 Using advanced tuning options If your jobs have CPU-intensive and memory-intensive operations, you can use the following advanced tuning features to improve performance: · Parallel processes--Individual work flows and data flows can execute in parallel if you do not connect them in the Designer workspace. 30 2010-12-02 Tuning Overview · Parallel threads--The software supports partitioned source tables, partitioned target tables, and degree of parallelism. These options allow you to control the number of instances for a source, target, and transform that can run in parallel within a data flow. Each instance runs as a separate thread and can run on a separate CPU. Server groups and distribution levels--You can group Job Servers on different computers into a logical component called a server group.

A server group automatically measures resource availability on each Job Server in the group and distributes scheduled batch jobs to the computer with the lightest load at runtime. This functionality also provides a hot backup method. If one Job Server in a server group is down, another Job Server in the group processes the job. You can distribute the execution of data flows or sub data flows within a batch job across multiple Job Servers within a Server Group to better balance resource-intensive operations. · Related Topics · Using Parallel Execution · Management Console Guide: Server Groups · Using grid computing to distribute data flow execution 31 2010-12-02 Tuning Overview 32 2010-12-02 Maximizing Push-Down Operations Maximizing Push-Down Operations For SQL sources and targets, SAP BusinessObjects Data Services creates database-specific SQL statements based on the data flow diagrams in a job. The software generates SQL SELECT statements to retrieve the data from source databases. To optimize performance, the software pushes down as many SELECT operations as possible to the source database and combines as many operations as possible into one request to the database. It can push down SELECT operations such as joins, Group By, and common functions such as decode and string functions. Data flow design influences the number of operations that the software can push to the database. Before running a job, you can view the SQL that is generated and adjust your design to maximize the SQL that is pushed down to improve performance.

You can use database links and the Data_Transfer transform to pushdown more operations. 5.1 Push-down operations By pushing down operations to the source database, Data Services reduces the number of rows and operations that the engine must retrieve and process, which improves performance.



[You're reading an excerpt. Click here to read official BUSINESS OBJECTS DATA SERVICES XI 4.0 user guide](http://yourpdfguides.com/dref/3594853)
<http://yourpdfguides.com/dref/3594853>

When determining which operations to push to the database, Data Services examines the database and its environment. 5.1.1 Full push-down operations The Optimizer always first tries to do a full push-down operation. A full push-down operation is when all transform operations can be pushed down to the databases and the data streams directly from the source database to the target database. SAP BusinessObjects Data Services sends SQL INSERT INTO.. . SELECT statements to the target database where SELECT retrieves data from the source. The software does a full push-down operation to the source and target databases when the following conditions are met: . . . All of the operations between the source table and target table can be pushed down. The source and target tables are from the same datastore or they are in datastores that have a database link defined between them. To enable a full push-down from the source to the target, you can also use the following features: 33 2010-12-02 Maximizing Push-Down Operations . . . Data_Transfer transform Database links For database targets that support the Allow Merge option, when all other operations in the data flow can be pushed down to the source database, the auto-correct loading operation may also be pushed down for a full push-down operation to the target.

The software sends an SQL MERGE INTO target statement that implements the Ignore columns with value and Ignore columns with null options. 5.1.2 Partial push-down operations When a full push-down operation is not possible, SAP BusinessObjects Data Services still pushes down the SELECT statement to the source database. Operations within the SELECT statement that the software can push to the database include: Aggregations -- Aggregate functions, typically used with a Group by statement, always produce a data set smaller than or the same size as the original data set. Distinct rows -- When you select Distinct rows from the Select tab in the query editor, the software will only output unique rows. Filtering -- Filtering can produce a data set smaller than or equal to the original data set. Joins -- Joins typically produce a data set smaller than or similar in size to the original tables. The software can push down joins when either of the following conditions exist: The source tables are in the same datastore The source tables are in datastores that have a database link defined between them Ordering -- Ordering does not affect data-set size. The software can efficiently sort data sets that fit in memory. It is recommended that you push down the Order By for very large data sets. Projection -- Projection is the subset of columns that you map on the Mapping tab in the query editor. Projection normally produces a smaller data set because it only returns columns needed by subsequent operations in a data flow. Functions -- Most functions that have equivalents in the underlying database are appropriately translated. These functions include decode, aggregation, and string functions.

5.1.3 Operations that cannot be pushed down SAP BusinessObjects Data Services cannot push some transform operations to the database. For example: . . . Expressions that include functions that do not have database correspondents 34 2010-12-02 Maximizing Push-Down Operations Load operations that contain triggers Transforms other than Query Joins between sources that are on different database servers that do not have database links defined between them. Similarly, the software cannot always combine operations into single requests. For example, when a stored procedure contains a COMMIT statement or does not return a value, the software cannot combine the stored procedure SQL with the SQL for other operations in a query. The software can only push operations supported by the DBMS down to that DBMS. Therefore, for best performance, try not to intersperse SAP BusinessObjects Data Services transforms among operations that can be pushed down to the database. 5.2 Push-down examples The following are typical push-down scenarios.

5.2.1 Collapsing transforms to push down operations example When determining how to push operations to the database, SAP BusinessObjects Data Services first collapses all the transforms into the minimum set of transformations expressed in terms of the source table columns. Next, the software pushes all possible operations on tables of the same database down to that DBMS. For example, the following data flow extracts rows from a single source table. 35 2010-12-02 Maximizing Push-Down Operations The first query selects only the rows in the source where column A contains a value greater than 100. The second query refines the extraction further, reducing the number of columns returned and further reducing the qualifying rows. The software collapses the two queries into a single command for the DBMS to execute. The following command uses AND to combine the WHERE clauses from the two queries: SELECT A, MAX(B), C FROM source WHERE A > 100 AND B = C GROUP BY A, C The software can push down all the operations in this SELECT statement to the source DBMS. 5.

2.2 Full push down from the source to the target example If the source and target are in the same datastore, the software can do a full push-down operation where the INSERT into the target uses a SELECT from the source. In the sample data flow in scenario 1, a full push down passes the following statement to the database: INSERT INTO target (A, B, C) SELECT A, MAX(B), C FROM source WHERE A > 100 AND B = C GROUP BY A, C If the source and target are not in the same datastore, the software can also do a full push-down operation if you use one of the following features: . . . Add a Data_Transfer transform before the target. . . Define a database link between the two datastores. 5.2.3 Full push down for auto correct load to the target example For supported databases, if you enable the Auto correct load and Allow Merge options, the Optimizer may be able to do a full push-down operation where the SQL statement is a MERGE into the target with a SELECT from the source. In order for the Allow Merge option to generate a MERGE statement, the primary key of the source table must be a subset of the primary key of the target table and the source row must be unique on the target key. In other words, there cannot be duplicate rows in the source data. If this condition is not met, the Optimizer pushes down the operation using a database-specific method to identify, update, and insert rows into the target table.

For example, suppose you have a data flow where the source and target tables are in the same datastore and the Auto correct load and Allow Merge options are set to Yes. 36 2010-12-02 Maximizing Push-Down Operations The push-down operation passes the following statement to an Oracle database: MERGE INTO "ODS".



[You're reading an excerpt. Click here to read official BUSINESS OBJECTS DATA SERVICES XI 4.0 user guide](http://yourpdfguides.com/dref/3594853)
<http://yourpdfguides.com/dref/3594853>

"TARGET" s USING SELECT "SOURCE"."A" A , "SOURCE"."B" B , "SOURCE"."C" C FROM "ODS"."SOURCE" "SOURCE")n ON ((s.A = n.A)) WHEN MATCHED THEN UPDATE SET s."B" = n.

B, s."C" = n.C WHEN NOT MATCHED THEN INSERT /*+ APPEND */(s."A", s."B", s."C") VALUES (n.A , n.B , n.C) Similar statements are used for other supported databases. 5.

2.4 Partial push down to the source example If the data flow contains operations that cannot be passed to the DBMS, the software optimizes the transformation differently than the previous two scenarios. For example, if Query1 called $\text{func}(A) > 100$, where func is a SAP BusinessObjects Data Services custom function, then the software generates two commands: · The first query becomes the following command which the source DBMS executes: `SELECT A, B, C FROM source WHERE B = C` · The second query becomes the following command which SAP BusinessObjects Data Services executes because func cannot be pushed to the database: `SELECT A, MAX(B), C FROM Query1 WHERE $\text{func}(A) > 100$ GROUP BY A, C` 5.2.5 Push-down of SQL join example If the tables to be joined in a query meet the requirements for a push-down operation, then the entire query is pushed down to the DBMS. To confirm that the query will be pushed down, look at the Optimized SQL. If the query shows a single SELECT statement, then it will be pushed down. For example, in the data flow shown below, the Department and Employee tables are joined with an inner join and then the result of that join is joined with left outer join to the Bonus table. 37 2010-12-02 Maximizing Push-Down Operations The resulting Optimized SQL contains a single select statement and the entire query is pushed down to the DBMS: `SELECT DEPARTMENT.DEPTID, DEPARTMENT.`

`DEPARTMENT, EMPLOYEE.LASTNAME, BONUS.BONUS FROM (DEPARTMENT INNER JOIN EMPLOYEE (ON DEPARTMENT.DEPTID=EMPLOYEE.DEPTID)) LEFT OUTER JOIN BONUS ON (EMPLOYEE.EMPID = BONUS.EMPID)` Related Topics · To view SQL · Maximizing Push-Down Operations · Reference Guide: Transforms, Query, Joins in the Query transform 5.3 To view SQL Before running a job, you can view the SQL code that SAP BusinessObjects Data Services generates for table sources in data flows. By examining the SQL code, you can verify that the software generates the commands you expect. If necessary, you can alter your design to improve the data flow.

1. Validate and save data flows. 2. Open a data flow in the workspace. 3.

Select Display Optimized SQL from the Validation menu. Alternately, you can right-click a data flow in the object library and select Display Optimized SQL. The Optimize SQL window opens and shows a list of datastores and the optimized SQL code for the selected datastore. By default, the Optimize SQL window selects the first datastore. The software only shows the SELECT generated for table sources and INSERT INTO.

.. SELECT for targets. It does not show the SQL generated for SQL sources that are not table sources, such as: · Lookup function 38 2010-12-02 Maximizing Push-Down Operations · · Key_generation function Key_Generation transform Table_Comparison transform 4. Select a name from the list of datastores on the left to view the SQL that this data flow applies against the corresponding database or application. The following example shows the optimized SQL for the second datastore which illustrates a full push-down operation (INSERT INTO... SELECT). This data flows uses a Data_Transfer transform to create a transfer table that the software loads directly into the target.

`INSERT INTO "DBO"."ORDER_AGG" ("SHIPCOUNTRY","SHIPREGION", "SALES_AGG") SELECT "TS_Query_Lookup"."SHIPCOUNTRY", "TS_Query_Lookup"."SHIPREGION", sum("TS_Query_Lookup"."SALES") FROM "DBO"."TRANS2""TS_Query_Lookup" GROUP BY "TS_Query_Lookup"."SHIPCOUNTRY", "TS_Query_Lookup"."SHIPREGION"` In the Optimized SQL window you can: · · Use the Find button to perform a search on the SQL displayed. Use the Save As button to save the text as a .sql file.

If you try to use the Display Optimized SQL command when there are no SQL sources in your data flow, the software alerts you. Examples of non-SQL sources include: · · Message sources File sources IDoc sources If a data flow is not valid when you click the Display Optimized SQL option, the software alerts you. Note: The Optimized SQL window displays the existing SQL statement in the repository. If you changed your data flow, save it so that the Optimized SQL window displays your current SQL statement. 5.

4 Data_Transfer transform for push-down operations Use the Data_Transfer transform to move data from a source or from another transform into the target datastore and enable a full push-down operation (INSERT INTO... SELECT) to the target. You can use the Data_Transfer transform to push-down resource-intensive operations that occur anywhere within a data flow to the database.

Resource-intensive operations include joins, GROUP BY, ORDER BY, and DISTINCT. 5.4.1 Push down an operation after a blocking operation example 39 2010-12-02 Maximizing Push-Down Operations You can place a Data_Transfer transform after a blocking operation to enable SAP BusinessObjects Data Services to push down a subsequent operation. A blocking operation is an operation that the software cannot push down to the database, and prevents ("blocks") operations after it from being pushed down. For example, you might have a data flow that groups sales order records by country and region, and sums the sales amounts to find which regions are generating the most revenue. The following diagram shows that the data flow contains a Pivot transform to obtain orders by Customer ID, a Query transform that contains a lookup_ext function to obtain sales subtotals, and another Query transform to group the results by country and region. Because the Pivot transform and the lookup_ext function are before the query with the GROUP BY clause, the software cannot push down the GROUP BY operation. The following Optimize SQL window shows the SELECT statement that the software pushes down to the source database: However, if you add a Data_Transfer transform before the second Query transform and specify a transfer table in the same datastore as the target table, the software can push down the GROUP BY operation. 40 2010-12-02 Maximizing Push-Down Operations The following Data_Transfer Editor window shows that the transfer type is table and the transfer table is in the same datastore as the target table.

The following Optimize SQL window shows that the software pushed down the GROUP BY to the transfer table TRANS2.



[You're reading an excerpt. Click here to read official BUSINESS OBJECTS DATA SERVICES XI 4.0 user guide](http://yourpdfguides.com/dref/3594853)
<http://yourpdfguides.com/dref/3594853>

41 2010-12-02 Maximizing Push-Down Operations Related Topics · Operations that cannot be pushed down 5.4.2 Using Data_Transfer tables to speed up auto correct loads example Auto correct loading ensures that the same row is not duplicated in a target table, which is useful for data recovery operations. However, an auto correct load prevents a full push-down operation from the source to the target when the source and target are in different datastores. For large loads using database targets that support the Allow Merge option for auto correct load, you can add a Data_Transfer transform before the target to enable a full push-down from the source to the target. In order for the Allow Merge option to generate a MERGE statement: · · the primary key of the source table must be a subset of the primary key of the target table the source row must be unique on the target key In other words, there cannot be duplicate rows in the source data. If this condition is not met, the Optimizer pushes down the operation using a database-specific method to identify, update, and insert rows into the target table. If the MERGE statement can be used, SAP BusinessObjects Data Services generates an SQL MERGE INTO target statement that implements the Ignore columns with value value (if a value is specified in the target transform editor) and the Ignore columns with null Yes/No setting. For example, suppose you create a data flow that loads sales orders into an Oracle target table which is in a different datastore from the source.

For this data flow, the Auto correct load option is active and set to Yes, and the Ignore columns with null and Allow merge options are also active. 42 2010-12-02 Maximizing Push-Down Operations The SELECT statement that the software pushes down to the source database would look like the following (as it would appear in the Optimize SQL window). SELECT "ODS_SALESORDER"."SALES_ORDER_NUMBER" , "ODS_SALESORDER"."ORDER_DATE" , "ODS_SALESORDER".

"CUST_ID" FROM "ODS"."ODS_SALESORDER""ODS_SALESORDER" When you add a Data_Transfer transform before the target and specify a transfer table in the same datastore as the target, the software can push down the auto correct load operation. The following MERGE statement is what the software pushes down to the Oracle target (as it appears in the Optimize SQL window). MERGE INTO "TARGET"."AUTO_CORRECT_LOAD2_TARGET" s USING (SELECT "AUTOLOADTRANSFER".

```
"SALES_ORDER_NUMBER" SALES_ORDER_NUMBER, "AUTOLOADTRANSFER"."ORDER_DATE" ORDER_DATE,
"AUTOLOADTRANSFER"."CUST_ID" CUST_ID FROM "TARGET"."AUTOLOADTRANSFER" "AUTOLOADTRANSFER") n ON
((s.SALES_ORDER_NUMBER=n.SALES_ORDRE_NUMBER00 WHEN MATCHED THEN UPDATE SET
s."ORDER_DATE"=nvl(n.ORDER_DATE,s."ORDER_DATE"), s."CUST_ID"=nbl(n.
```

```
CUST_ID,s."CUST_ID" WHEN NOT MATCHED THEN INSERT(s."SALES_ORDER_NUMBER",s."ORDER_DATE",s."CUST_ID")
VALUES(n.SALES_ORDRE_NUMBER,n.ORDRE_DATE,n.CUSTID) Similar statements are used for other supported databases. 5.5 Database link support
for push-down operations across datastores Various database vendors support one-way communication paths from one database server to another.
SAP BusinessObjects Data Services refers to communication paths between databases as database links. The datastores in a database link relationship are
called linked datastores. The software uses linked datastores to enhance its performance by pushing down operations to a target database using a target
datastore. Pushing down operations to a database not only reduces the amount of information that needs to be transferred between the databases and SAP
BusinessObjects Data Services but also allows the software to take advantage of the various DMBS capabilities, such as various join algorithms. With support
for database links, the software pushes processing down from different datastores, which can also refer to the same or different database type.
```

Linked datastores allow a one-way path for data. For example, if you import a database link from target database B and link datastore B to datastore A, the software pushes the load operation down to database B, not to database A. This section contains the following topics: · · · Software support Example of push-down with linked datastores Generated SQL statements Tuning performance at the data flow or Job Server level 43 2010-12-02 Maximizing Push-Down Operations Related Topics · Designer Guide: Datastores, Linked datastores 5.5.1 Software support SAP BusinessObjects Data Services supports push-down operations using linked datastores on all Windows and Unix platforms.

It supports DB2, Oracle, and MS SQL server databases. 5.5.2 To take advantage of linked datastores 1. Create a database link on a database server that you intend to use as a target in a job. The following database software is required. See the Supported Platforms document for specific version numbers. · For DB2, use the DB2 Information Services (previously known as Relational Connect) software and make sure that the database user has privileges to create and drop a nickname. To end users and client applications, data sources appear as a single collective database in DB2. Users and applications interface with the database managed by the information server.

Therefore, configure an information server and then add the external data sources. DB2 uses nicknames to identify remote tables and views. See the DB2 database manuals for more information about how to create links for DB2 and non-DB2 servers. · For Oracle, use the Transparent Gateway for DB2 and MS SQL Server. See the Oracle database manuals for more information about how to create database links for Oracle and non-Oracle servers. · For MS SQL

Server, no special software is required. Microsoft SQL Server supports access to distributed data stored in multiple instances of SQL Server and heterogeneous data stored in various relational and non-relational data sources using an OLE database provider. SQL Server supports access to distributed or heterogeneous database sources in Transact-SQL statements by qualifying the data sources with the names of the linked server where the data sources exist. See the MS SQL Server database manuals for more information. 2.

Create a database datastore connection to your target database. 44 2010-12-02 Maximizing Push-Down Operations 5.5.3 Example of push-down with linked datastores Linked datastores enable a full push-down operation (INSERT INTO.

. SELECT) to the target if all the sources are linked with the target. The sources and target can be in datastores that use the same database type or different database types. The following diagram shows an example of a data flow that will take advantage of linked datastores: The dataflow joins three source tables from different database types: · · · ora_source.



[You're reading an excerpt. Click here to read official BUSINESS OBJECTS DATA SERVICES XI 4.0 user guide](http://yourpdfguides.com/dref/3594853)
<http://yourpdfguides.com/dref/3594853>