ONESOURCE™ REPORTING

SIZING GUIDE

6.8.X.X

Document Version 1

THOMSON REUTERS®

COPYRIGHT NOTICE

DOCUMENT HISTORY

VERSION NUMBER	VERSION DATE	SUMMARY
V1	October 10, 2020	Created first version of this guide for ONESOURCE Indirect Tax Reporting 6.8.0.0

TABLE OF CONTENTS

INTRODUCTION

This document discusses various technical aspects involved with determining how to configure ONESOURCE Indirect Tax Reporting for optimal performance. Due to the hardware, network, operating system, and usage pattern differences that may exist in your environment, none of these guidelines are guaranteed and should be viewed as suggestions only. While we strive to be as accurate as possible, your actual results may differ. This document also cannot cover every possible permutation of configuration options. For assistance with complex configuration options, contact Thomson Reuters Professional Services.

RESOURCES

Several resources help you become familiar with ONESOURCE Indirect Tax Reporting and master its features. Help is installed with the application. All documents are posted on the ONESOURCE Customer Center. To locate documents specific to your application, enter the search term "Reporting documentation."

REPORTING RESOURCES	
Resource	**Description**
Help	This Help system gives assistance within Reporting. Use Help after Reporting is installed and configured.
Installation Guide	This guide is intended for technical users and contains complete details about how to install and configure Reporting.
Platform Support	This describes the combinations of operating systems, databases, and application servers on which Reporting operates.
Product Support Lifecycle	This lists the end-of-life dates for products in the ONESOURCE Indirect Tax Suite.
Upgrade Guide	This guide describes the procedures for upgrading an instance of Reporting and refers to configuration information in the *Installation Guide*.
Customization Guide	This guide is intended for technical users. It describes types of customization and shows examples.
Data Dictionary	This resource is intended for technical users. It contains a list of all the fields in the Reporting database.
Sizing Guide	This guide is intended for technical users. It contains an architectural overview and discusses components, database, and server sizing.

Still can't find what you're looking for? Try these additional resources:

ONESOURCE RESOURCES	
Resource	**Description**
ONESOURCE Customer Center https://tax.thomsonreuters.com/support/onesource/customer-center/	Search for answers in the Knowledge Base, enter product support tickets, and track support ticket history for you and your organization.
Indirect Tax Customer Center https://customercenter.sabrix.com/	Download ONESOURCE Indirect Tax software.
Other ONESOURCE Indirect Tax Products and Services https://tax.thomsonreuters.com/products/brands/onesource/indirect-tax/	Browse descriptions of other ONESOURCE Indirect Tax products and services.
Documentation Feedback onesource.indirect.tax.fb@thomsonreuters.com	Send feedback about ONESOURCE Indirect Tax documentation.

ARCHITECTURAL OVERVIEW

This section describes system architecture:

Reporting is a multi-tier architecture that allows for both horizontal and vertical scaling. You can combine each application tier onto a single server or install them onto separate servers. You can cluster each tier as needed to scale with your business needs.

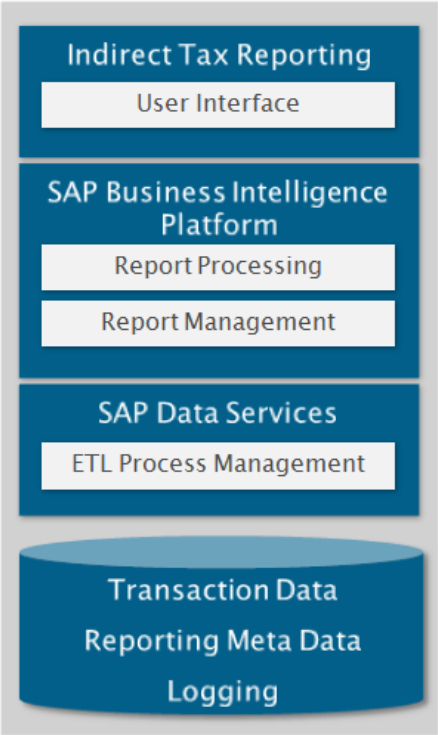
You need to answer several questions to determine the optimal Reporting configuration.

CONSIDERATIONS

Some considerations that influence the final configuration include:

- How many Reporting users will you have?
- How many reports will you run concurrently?
- Will users run reports for one company/division or several?
- How often will you run the ETL process?
- How much tax history will you initially load into Reporting?
- How many transactions will be processed with each ETL?
- How much time have you allocated for your ETL processing window?

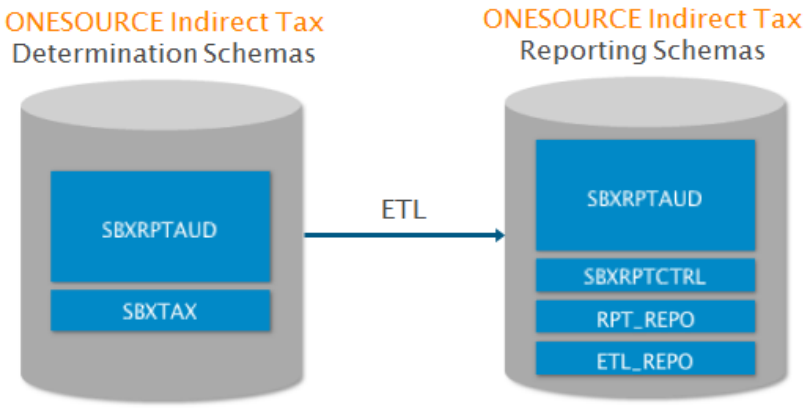
COMPONENT DESCRIPTIONS



This section illustrates the architecture of the Reporting configuration.

- Database architecture
- ETL architecture
- Report processing architecture
- Web application tier

Database Architecture

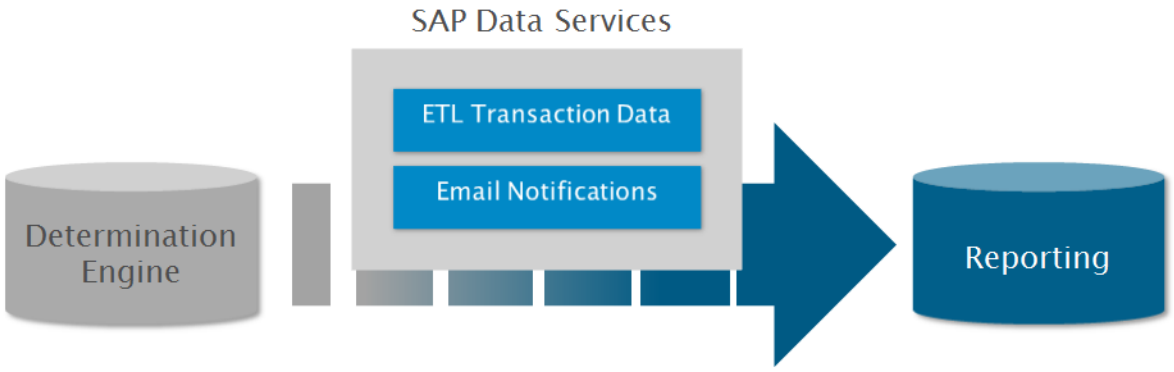


The database tier for Reporting stores the report data and processes the queries used to build the reports. The database level consists of your source data that originates in ONESOURCE Indirect Tax Determination the schemas used to hold the Reporting data, and metadata defining the ETL process and report definitions.

Each of the schemas shown in the diagram can be stored on the same or separate database instances. However, we recommend separating your source Determination data from your Reporting data for better performance. The database configuration affects performance as every report requested will query the database for data.

Depending on the size of your data and the date range of your report, the database processing can be significant. You can use your database vendor performance tools to evaluate throughput and optimize your database configuration. These techniques are beyond the scope of this document.

Data Management Architecture



The ETL tier is responsible for:

- Reading the source data.
- Optimizing the data for reports.
- Loading the Reporting data.

This tier is built on SAP Data Services. The ETL Job Server attempts to use all of the available resources on the server during processing. To optimize performance, we recommend hosting the ETL tier on its own physical server and configuring the ETL tier with high-speed connections to the source and Reporting databases.

The ETL tier can also be clustered across multiple machines by using Server Groups. See ETL Clustering with Server Groups (page 33) for additional information. Server Groups allow the ETL Job Server to distribute whole processes, a step of the process, or even sub-steps onto multiple machines to maximize the overall computing capacity. You should consider Server Groups if the amount of data moved in each ETL process is large or your ETL processing window is small.

Report Processing Architecture

The report processing tier is responsible for generating the reports. It uses components of SAP BusinessObjects BI Platform. This tier:

- Manages the input and output file requests to the disk storage.
- Submits the report requests.
- Sends the SQL to the database.
- Builds the reports.
- Takes requests from the user interface.
- Sends the completed report back to the user interface.

This tier also manages a disk cache to hold report data, so it is critical that you have sufficient disk space. If you have a large amount of data that will be included in your reports, consider using a Storage Area Network (SAN) or separate file storage. Be sure that the connection between the Reporting tier and the disk storage is high speed as a slow network will diminish report processing performance.

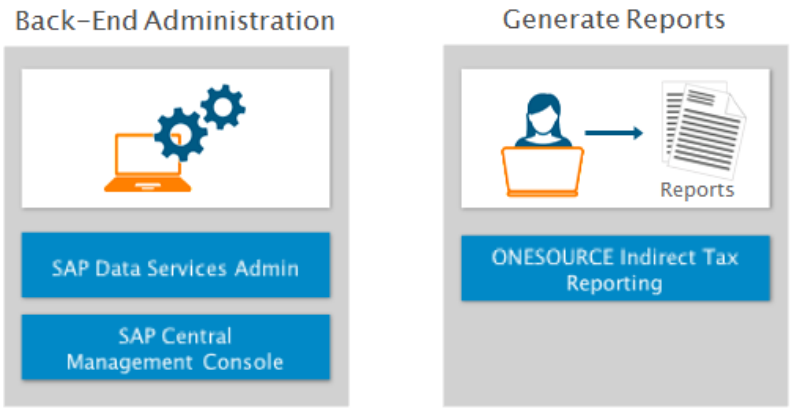
These BI Platform services require additional disk space:

- Crystal Reports cache
- Input file repository
- Output file repository

This tier has several services that cooperate to generate each report. Each service can be tuned individually to best fit your requirements. See Configuring Report Processing Services (page 12) for more information about each service.

The report processing tier can be combined on a single physical server with the other tiers. However, for optimal performance we recommend that you place this tier on its own server. You might consider increasing your services or clustering your report processing tier across multiple machines if you have a large amount of data or need to process many concurrent reports.

Web Application Tier



The web application tier provides the user interface for Reporting and two administrative consoles. This tier uses your web application server (for example, Tomcat, JBoss, or WebLogic). This tier does not process reports and makes minimal database requests to verify security.

Reporting communicates with the reporting tier to provide you with a list of available reports and to request execution of specific reports. Reporting then receives the finished report and renders it for viewing. Reporting also has a set of supporting Web Services.

Two additional applications in the web application tier administer the ETL and Reporting tiers. These applications are not visible to end users and are rarely used after the initial installation and scheduling of ETL processes. These administrative applications are light-weight and perform no processing on the web application server.

All of the applications can be put into a clustered environment for high availability or to increase the number of users who can concurrently access the system. Follow the instructions for your application server to make these changes.

DATABASE SIZING

This section describes the schema and shows how to plan for growth.

TABLE DESCRIPTIONS AND ASSUMPTIONS

The Reporting tables can be divided into two categories. In this document, we assume that there are two tablespaces holding these two categories of data:

- Slow-growing control tables (SBXRPTCTRL)
- Fast-growing data tables (SBXRPTAUD)

Control tables include report definition metadata, ETL definitions, system configurations, and report logging. Fast-growing data includes all of the transaction data staged for reporting.

The SBXRPTCTRL tables will grow at roughly the same rate regardless of your Audit database size. Because these tables capture history, the primary factors contributing to the growth of the tables are the number of system users and the number of reports processed.

SIZING RECOMMENDATIONS FOR REPORT CONTROL DATA

The installation process creates the report control tables and populates core data used by Reporting. Since additional data will be written to this table as part of report processing, proper sizing requires understanding the growth rate. The report control tables that grow most rapidly are Report logging and ETL logging.

Report logging can be sized as follows:

TABLE	DESCRIPTION	SIZE PER ROW (BYTES)	AVERAGE ROWS PER REPORT INSTANCE	TOTAL AVERAGE SIZE PER REPORT INSTANCE
RC_REPORT_ LOG	Records each instance of the report	806	1	806
RC_REPORT_ PARAM_ VALUES_LOG	Records all the parameter values selected for each report request	2102	8	16816

TABLE	DESCRIPTION	SIZE PER ROW (BYTES)	AVERAGE ROWS PER REPORT INSTANCE	TOTAL AVERAGE SIZE PER REPORT INSTANCE
RC_REPORT_ PARAMETERS_ LOG	Records the parameter names selected for each report request	312	8	2496
Total				20118

Using the table above you can calculate the growth rate/year for Report logging as:

# of users * # of reports run per month * 20118 * 12 = annual report log growth rate in bytes.

For example, if you have 10 Reporting Users who each run approximately 30 reports per month you would use the formula:

10 * 30 * 20118 *12 = 108424800 bytes or about 70 MB per year expected growth.

ETL Logging can be sized as follows:

TABLE	DESCRIPTION	SIZE PER ROW
RS_BATCH_ CONTROL	Records each instance of the ETL Process	511
RS_JOB_LOG	Records count and amount total for each company processed by the ETL Process	1560

Using the chart above you can calculate the growth rate/year for ETL logging as:

# of ETL runs per day * 365 *(511 + (# of companies * 1560)) = annual ETL log growth rate in bytes.

For example, if you run the ETL two times per day and you have five companies you would use the formula:

2 * 365 * (511 + (5 * 1560)) = 6067030 bytes or about 6 MB per year expected growth.

Your total annual growth rate would be your ETL log growth rate + report log growth rate. Using the examples above this is 70 + 6 = 76 MB per year.

We recommend starting with a tablespace size of 300 MB for Report Control to allow room for growth. However, if your annual growth rate is larger than this minimum, you should create a larger tablespace. We recommend allowing your tablespace to grow automatically to avoid out of space errors later.

SIZING RECOMMENDATIONS FOR REPORT DATA

The tables used to hold the report data will grow at the same rate of time as the Determination audit database and will be slightly larger in size. Set the initial size of the SBXRPTAUD database between 110-150% the size of SBXAUD. We recommend that you allow this tablespace to grow automatically to avoid out-of-space errors.

SERVER SIZING

Client Tier	No minimum requirements imposed by Reporting
Middle Tier*	• RAM: 16GB • CPU: Quad-core processor with minimum 2GHz ** • Disk Space: 30GB
Database Tier*	• RAM: 2GB • CPU: No minimum requirement • Disk Space: 2GB

*These represent the minimum configuration, dictated as minimums by the SAP components running Reporting, and may not provide the best performance. Increasing server memory and processor speed will improve performance.

**For Windows, a Pentium 4-class processor is recommended. For Linux, only AMD/Intel chipsets are supported.

The report processing will create various temporary files on the middle tier server. These files may be quite large, depending on the number of reports run and the amount of data returned for each report. For example, a US Document Reconciliation report returning 5000 lines is approximately 2 MB. A configurable parameter in the Reporting system lets you define the cache size. By default, the cache size is established at 250 MB. A larger cache size (using more disk space) may be necessary to process large numbers of reports or reports that are especially complex. See Configuring Report Processing Services (page 12) for more information.

CONFIGURING REPORT PROCESSING SERVICES

The processing of reporting data into reports is handled by several services running on your application server, each of which may be configured. Configurations include time-out settings, report size limitations, cache settings, directory locations, and port assignments, among others.

You may see other servers in your environment. Those listed here are used directly by Reporting. You can improve performance by freeing up server capacity and stopping unnecessary services.

SERVICE NAME	DESCRIPTION
Crystal Reports Cache Server	The Crystal Reports Cache Server intercepts report requests sent from clients to the page server. If the cache server cannot fulfill the request with a cached report page, it passes the request on to the page server, which runs the report and returns the results. The cache server then caches the report page for future use, and sends the report to the viewer.
Adaptive Job Server	The Adaptive job server processes scheduled actions on objects at the request of the CMS.
Crystal Reports 2013 Processing Server	The Crystal Reports 2013 Processing Server is responsible for responding to page requests by processing reports and generating encapsulated page format (EPF) pages. The key benefit of EPF is that it supports page-on-demand access so that only the requested page is returned, instead of the entire report. This enhances performance and reduces unnecessary network traffic for large reports.
Crystal Reports 2013 Report Application Server	The Crystal Reports 2013 Report Application Server provides ad-hoc reporting capabilities that allow you to create and modify Crystal reports via the RAS Software Development Kit (SDK).
Input File Repository	The Input File Repository stores report and program objects that have been published to the system.
Output File Repository	The Output File Repository stores all of the report instances generated by the Reports Job Server.
Central Management Server	The Central Management Server (CMS) maintains a database of information about your SAP BusinessObjects BI Platform system. This is known as the CMS system database. All platform services are managed and controlled by the CMS. The CMS also manages access to the system file store where the physical documents are managed.

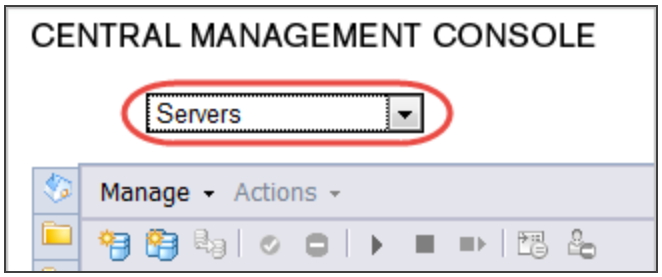
REVIEWING REPORT METRICS

The Servers management area of the CMC displays metrics about each server. For each service, the display includes information about the machine on which the service is running: name, operating system, total hard disk space, free hard disk space, total RAM, number of CPUs, and local time. The general information also includes the time the service started and the version number of the services.

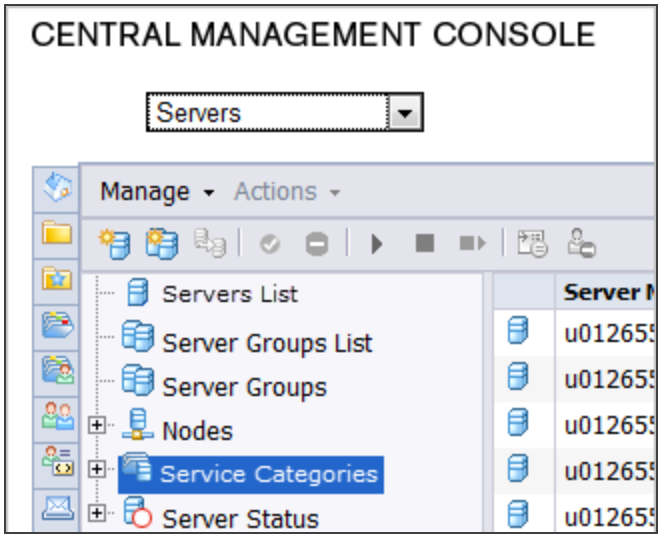
Review Report Metrics

This procedure shows you how to view the metrics for the Crystal Cache Server.

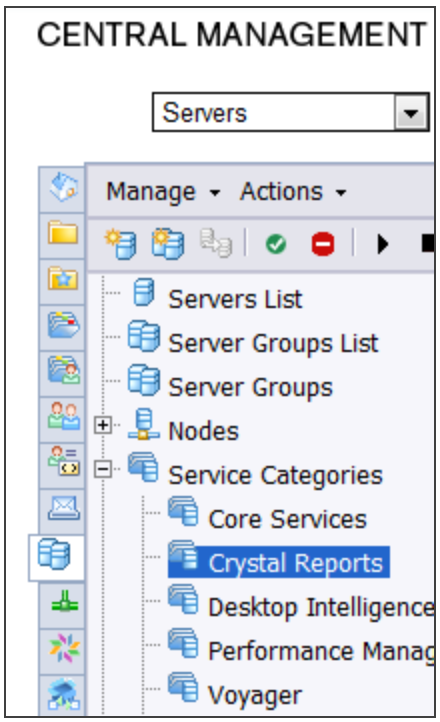
1. Open a browser and enter the CMC URL (http://<host>:<port>/BOE/CMC).
2. Log in to the CMC as **Administrator**.
3. Navigate to the list of services (called **Servers** in CMC.)



4. Select **Service Categories**.

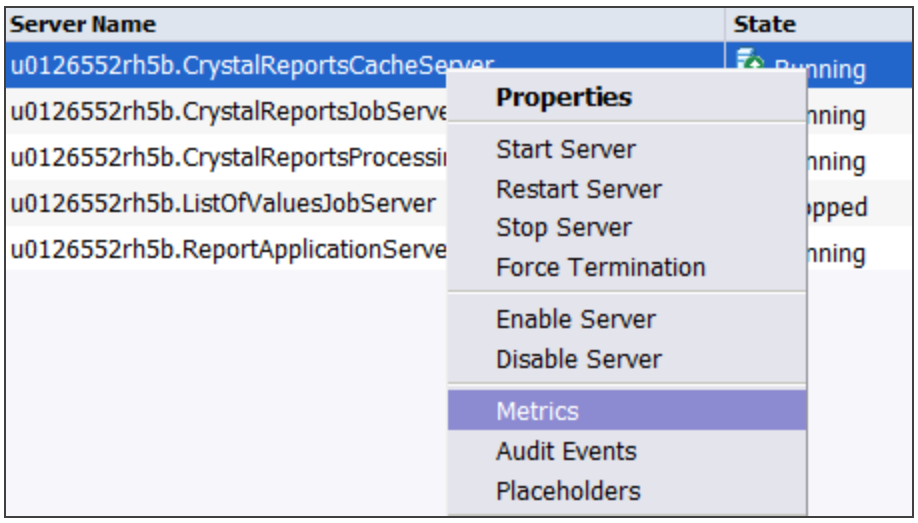


5. Select **Crystal Reports**.



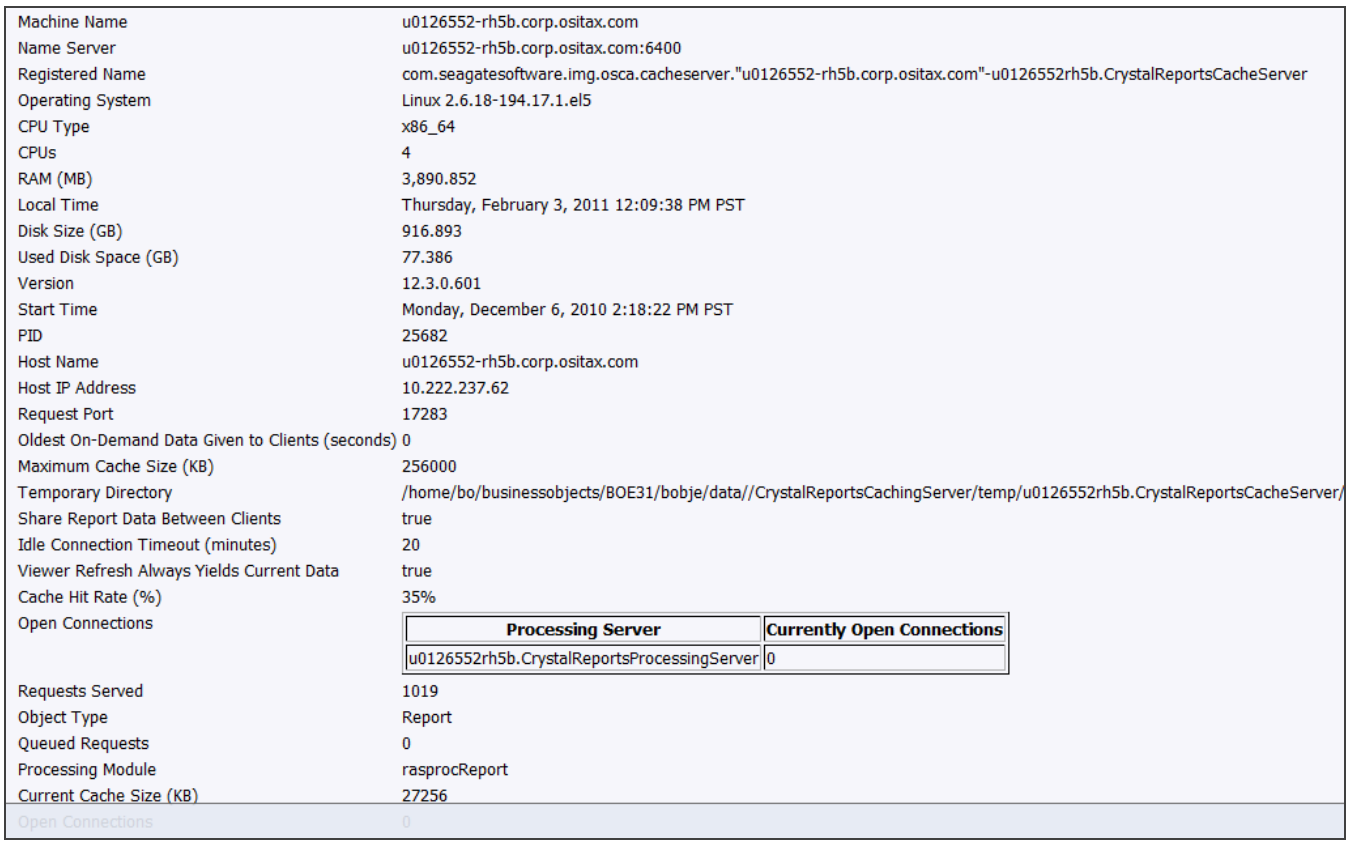
6. Locate and select the service you want to change (in this example, **Crystal Cache Server**).

7. Right-click and select **Metrics**.



There may be multiple instances of the service. If so, you will want to repeat this process for each of them.

8. You can now review the metrics, as shown in the service details below.



Machine Name	u0126552-rh5b.corp.ositax.com
Name Server	u0126552-rh5b.corp.ositax.com:6400
Registered Name	com.seagatesoftware.img.osca.cacheserver."u0126552-rh5b.corp.ositax.com"-u0126552rh5b.CrystalReportsCacheServer
Operating System	Linux 2.6.18-194.17.1.el5
CPU Type	x86_64
CPUs	4
RAM (MB)	3,890.852
Local Time	Thursday, February 3, 2011 12:09:38 PM PST
Disk Size (GB)	916.893
Used Disk Space (GB)	77.386
Version	12.3.0.601
Start Time	Monday, December 6, 2010 2:18:22 PM PST
PID	25682
Host Name	u0126552-rh5b.corp.ositax.com
Host IP Address	10.222.237.62
Request Port	17283
Oldest On-Demand Data Given to Clients (seconds)	0
Maximum Cache Size (KB)	256000
Temporary Directory	/home/bo/businessobjects/BOE31/bobje/data//CrystalReportsCachingServer/temp/u0126552rh5b.CrystalReportsCacheServer/
Share Report Data Between Clients	true
Idle Connection Timeout (minutes)	20
Viewer Refresh Always Yields Current Data	true
Cache Hit Rate (%)	35%
Open Connections	

Processing Server	Currently Open Connections
u0126552rh5b.CrystalReportsProcessingServer	0

Requests Served	1019
Object Type	Report
Queued Requests	0
Processing Module	rasprocReport
Current Cache Size (KB)	27256
Open Connections	0

Input and Output File Repository Server Metrics

For each File Repository Server, the **Metrics** tab provides the following information:

- The data sent and data written.
- The number of active files and active client connections.
- The total available disk space.

Central Management Server Metrics

For the CMS, the **Metrics** tab provides the following information:

- The number of jobs that are failed, pending, running, completed, or waiting
- The number of licenses
- CMS system database information

Adaptive Job Server Metrics

The **Metrics** tab of these job servers lists the following information:

- Location of its temporary files
- Processing mode
- Current number of jobs being processed
- Total number of requests received
- Total number of failed job creations
- Types of default destinations currently enabled

Crystal Reports Cache Server Metrics

For each Cache Server, the **Metrics** tab provides the following information:

- Number of bytes transferred
- Number of current connections
- Current cache size
- Number of requests served
- Cache hit rate
- Number of requests that are queued

The **Metrics** tab also displays the current values for the following settings, which can be changed on the **Properties** tab:

- Number of minutes before an idle job is closed
- If the database is accessed when a viewer's file (object) is refreshed
- Location of the cache files
- Maximum cache size
- Number of minutes between refreshes from the database

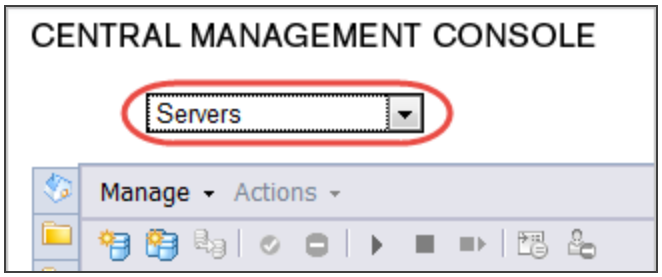
The **Metrics** tab also provides a table that lists the Processing Servers to which the Cache Server has connections, along with the number of connections made to each Processing Server.

CHANGING SERVICE SETTINGS

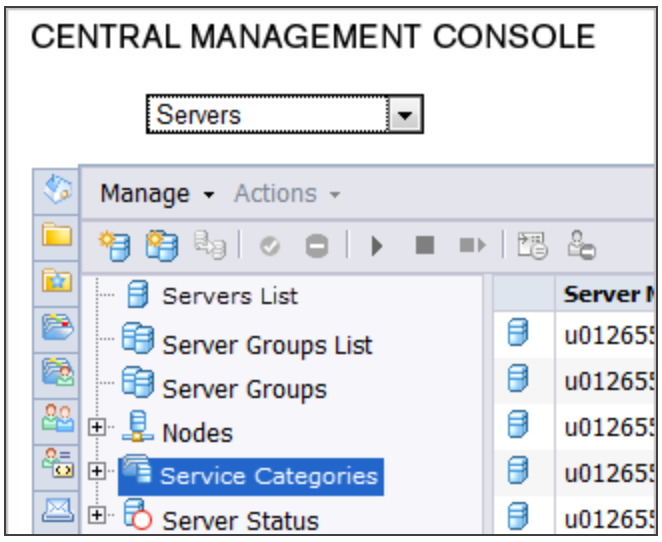
All changes to the Service settings are made in the Central Management Console (CMC), an administrative tool intended for use by your IT staff.

In this example, we change the default record limit in the Crystal Processing Server to unlimited.

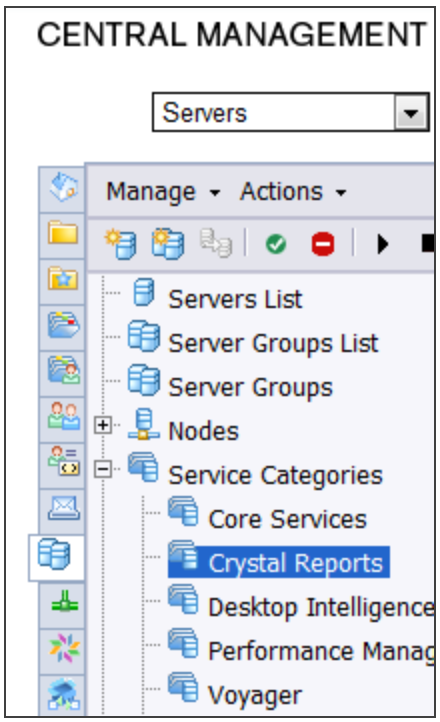
1. Open a browser and enter the CMC URL (http://<host>:<port>/BOE/CMC).
2. Log in to the CMC as **Administrator**.
3. Navigate to the list of services (called **Servers** in CMC.)



4. Select **Service Categories**.

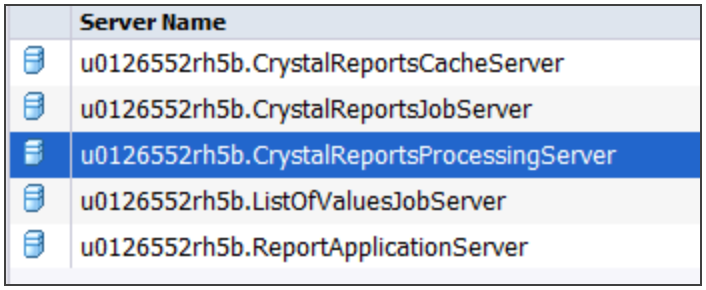


5. Select **Crystal Reports**.

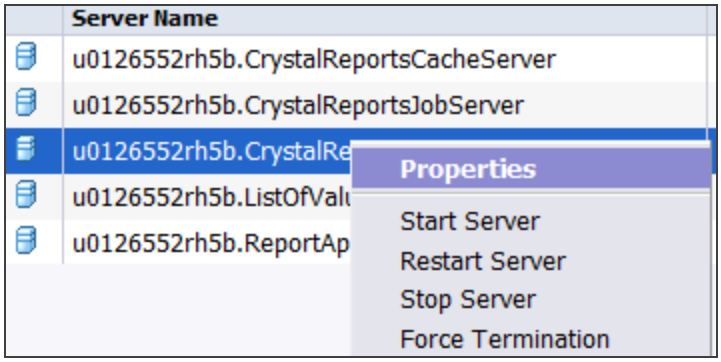


6. Locate the service you wish to change. In this example, select **Crystal Processing Server**.

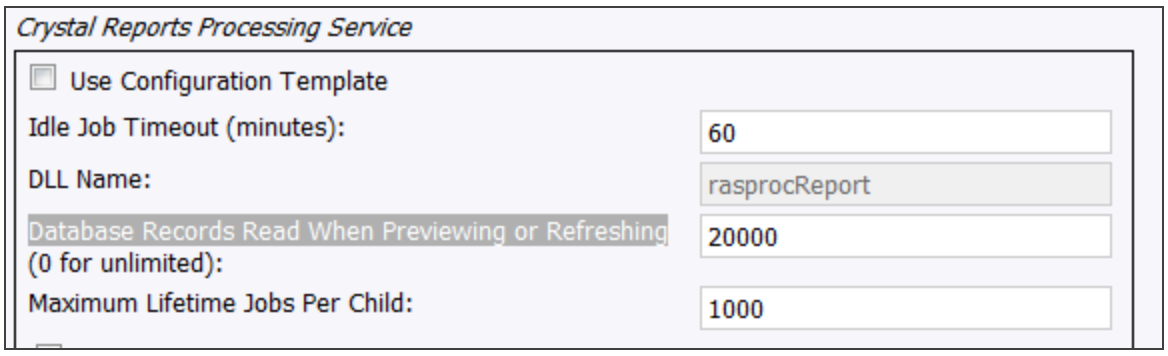
There may be multiple instances of the Crystal Processing Server. If so, repeat this process for each of them.



7. Right-click and select **Properties**.



8. Locate the property you want to change and make the appropriate changes. In this example, the value for **Database Records Read When Previewing or Refreshing**, was changed from **20000** to **0** to allow an unlimited number of records to be returned for each report.

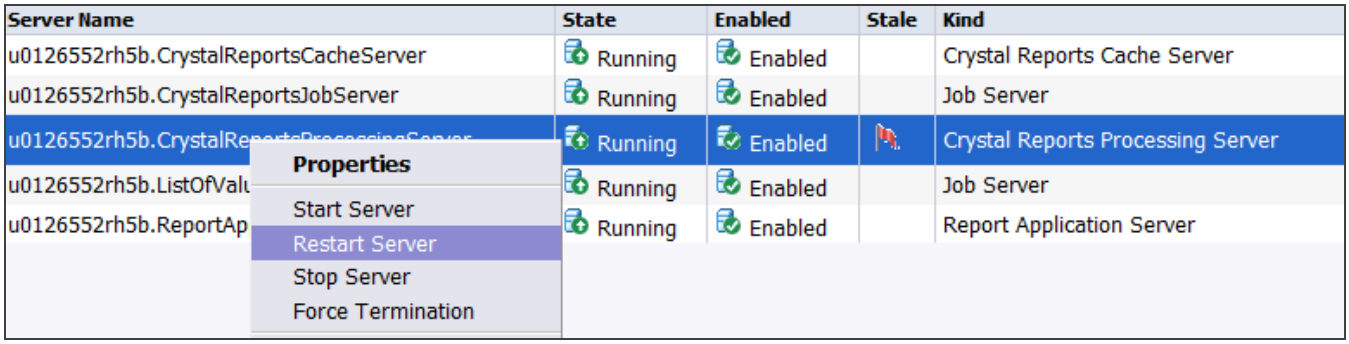


9. When you complete all changes, click **Save** and then **Close** at the bottom of the window to return to the service list.

10. If a restart is required, select the service from the list.

    If a restart of the service is required, a flag displays in the **Stale** column.

11. Right-click and select **Restart**.



CRYSTAL REPORTS SERVICES

Default values are intended to support average conditions which may not apply to your environment. We recommend that you start with the default values and evaluate the performance in your test environment, and then make changes as needed.

Adaptive Job Server Settings

When you modify any of these service settings, you must restart the service for the changes to take effect.

PROPERTY	DESCRIPTION	DEFAULT VALUES
Maximum Concurrent Jobs	Specifies the number of concurrent independent processes (child processes) that the service allows. You adjust the maximum number of jobs to suit your Reporting environment. The default setting is acceptable for most implementations. The ideal setting for your Reporting environment depends on your hardware configuration, database software, and Reporting requirements.	5

PROPERTY	DESCRIPTION	DEFAULT VALUES
Temporary Directory	Specifies the directory where temporary files are created when necessary.  You may encounter performance issues if this directory does not have adequate disk space.	%DefaultDataDir%

Crystal Reports Cache Server Settings

Any properties that apply to both Crystal Reports Cache Servers and Crystal Reports 2013 Processing Servers should be set to the same value. For example, if you set the Viewer Refresh Always Yields Current Data setting to TRUE on the Cache Server, you should set the same property to TRUE on the Processing Server.

 When you modify any of these service settings, you must restart the service for the changes to take effect.

PROPERTY	DESCRIPTION	DEFAULT VALUES
Viewer Refresh Always Yields Current Data	Specifies whether, when users explicitly refresh a report, all cached pages are ignored and new data is retrieved directly from the database.	TRUE
Share Report Data Between Clients	Specifies whether report data is shared between different clients.	TRUE
Idle Connection Timeout (minutes)	Specifies the amount of time, in minutes, that the Crystal Reports Cache Server waits for a request from an idle connection. There is generally no need to modify the default value.	20 minutes

PROPERTY	DESCRIPTION	DEFAULT VALUES
Oldest On-Demand Data Given to Clients (seconds)	Specifies the amount of time in seconds that the service uses cached data to meet requests from on-demand reports. If the service receives a request that can be met using data that was generated to meet a previous request, and the time elapsed since that data was generated is less than the value set here, then the service will reuse this data to meet the subsequent request. Reusing data in this way significantly improves system performance when multiple users need the same information. When setting this value consider how important it is that your users receive up-to-date data. If it is very important that all users receive fresh data you may need to disallow this kind of data reuse by setting the value to 0.	0 seconds
Maximum Cache Size (KB)	Specifies the amount of hard disk space (in KB) used to cache reports. A large cache size may be necessary if the service needs to handle large numbers of reports, or reports that are especially complex.	256000 KB
Cache Files Directory	Specifies the location of the cache file directory.	%DefaultDataDir%/Crystal ReportsCaching/temp

Crystal Reports/Crystal Reports 2013 Processing Server Settings

Properties that apply to both Crystal Reports Cache Servers and Crystal Reports 2013 Processing Servers should be set to the same value. For example, if you set the **Viewer Refresh Always Yields Current Data** setting to **TRUE** on the Cache Server, you should set the same property to **TRUE** on the Processing Server.

When you modify any of these service settings, you must restart the service for the changes to take effect.

PROPERTY	DESCRIPTION	DEFAULT VALUES
Idle Job Timeout (minutes)	Specifies the length of time, in minutes, that the Crystal Reports 2013 Processing Server waits between requests for a given job.	60 minutes

PROPERTY	DESCRIPTION	DEFAULT VALUES
Database Records Read When Previewing or Refreshing (0 for unlimited)	Specifies the maximum number of database records to read when the report is being previewed or refreshed. It allows you to limit the number of records that the database will return when a user runs a report. This setting is useful when you want to prevent users from running on-demand reports containing queries that return excessively large record sets. You may prefer to schedule such reports, both to make the reports available more quickly to users and to reduce the load on your database from these large queries.	20000
Maximum Lifetime Jobs Per Child	Specifies the maximum number of jobs that each child process can manage per lifetime.	1000
Viewer Refresh Always Yields Current Data	Specifies whether, when users explicitly refresh a report, all cached pages are ignored and new data is retrieved directly from the database.	TRUE
Share Report Data Between Clients	Specifies whether report data is shared between different clients.	TRUE
Idle Connection Timeout (minutes)	Specifies the amount of time, in minutes, that the Crystal Reports 2013 Processing Server waits for a request from an idle connection. There is generally no need to modify the default value.	20 minutes
Maximum Concurrent Jobs (0 for automatic)	Specifies the maximum number of independent jobs allowed to run concurrently on the Crystal Reports 2013 Processing Server. If the value of this property is set to “0”, the service applies a suitable value, based on the CPU and memory of the machine on which the service is running.	0

PROPERTY	DESCRIPTION	DEFAULT VALUES
Oldest On-Demand Data Given to Clients (seconds)	Specifies the amount of time in seconds that the service uses cached data to meet requests from on-demand reports. If the service receives a request that can be met using data that was generated to meet a previous request, and the time elapsed since that data was generated is less than the value set here, then the service will reuse this data to meet the subsequent request. Reusing data in this way significantly improves system performance when multiple users need the same information. When setting this value consider how important it is that your users receive up-to-date data. If it is very important that all users receive fresh data you may need to disallow this kind of data reuse by setting the value to 0.	0
Maximum Number of Prestarted Children	Specifies the maximum number of prestarted child processes that are allowed by the service. If this value is too low, the service creates child processes as soon as requests are made, and a user may experience latency. If this value is too high, system resources may be wasted by idle child processes.	1 child
Temporary Directory	Specifies the directory where temporary files are created when necessary.  You may encounter performance issues if this directory does not have adequate disk space.	%DefaultDataDir%/Crystal ReportsProcessingServer/temp
Allow Report Jobs to Stay Connected to the Database until the Report Job is Closed	Specifies whether the report job will remain connected to the database until the job is closed.	FALSE

Crystal Reports 2013 Report Application Server Settings

This service is used only by the Transaction Extract Report when it is being configured by the User Interface. Running the Transaction Report uses the Crystal Reports Processing Server.

When you modify any of these settings, you must restart the service for the changes to take effect.

PROPERTY	DESCRIPTION	DEFAULT VALUES
Allow Report Jobs to Stay Connected to the Database until the Report Job is Closed	Specifies whether the report job will remain connected to the database until the process has been executed.	FALSE
Browse Data Size (records)	Specifies the number of distinct records returned from the database when browsing through a particular field's values. The data is retrieved first from the client's cache - if it is available - and then from the service's cache. If the data is not in either cache, it is retrieved from the database.	100 records
Idle Connection Timeout (minutes)	Specifies the amount of time, in minutes, that the Crystal Reports 2013 Report Application Server (RAS) waits for requests from an idle client before timing out. Setting a value too low can cause a user's request to be closed prematurely, and setting a value that is too high can affect the service's scalability (for instance, if the ReportClientDocument object is not closed explicitly, the service will be waiting unnecessarily for an idle job to close).	30 minutes
Batch Size (records)	Specifies how many rows from the result set are returned by the database during each data transfer. For example, if 500 records are requested, and the Batch Size property is set to 100 records, the data will be returned in 5 separate batches of 100 rows. To improve the performance of your RAS, you must understand your network environment, database, and the type of requests in order to set the appropriate batch size.	100 records

PROPERTY	DESCRIPTION	DEFAULT VALUES
Number of database records to read when previewing or refreshing a report (-1 for unlimited)	Specifies the number of database records that will be read when viewing or refreshing a report. This setting limits the number of records that the service retrieves from the database when a user runs a query or report. This setting is useful when you want to prevent users from running on-demand reports that return excessively large record sets.	20000 records
Maximum Concurrent Report Jobs (0 for unlimited)	Specifies the maximum number of independent jobs allowed to run concurrently on the RAS.	75 jobs
Oldest on-demand data given to a client (minutes)	Specifies the amount of time, in minutes, an on-demand report will serve cached report data.	20 minutes

CORE REPORTS SERVICES

Core report services include the Input File Repository Server Settings and Output File Repository Server Settings.

Input File Repository Server Settings

PROPERTY	DESCRIPTION	DEFAULT VALUES
Maximum Retries for File Access	Specifies the number of times the server tries to access a file.	1

PROPERTY	DESCRIPTION	DEFAULT VALUES
Maximum Idle Time (minutes)	Specifies the length of time that the server waits before it closes inactive connections. Setting a value that is too low can cause your request to close prematurely. Setting a value that is too high can cause excessive consumption of system resources such as processing time and disk space.	10 minutes
Temporary Directory	Specifies the directory where temporary files are created when necessary.  You may encounter performance issues if this directory does not have adequate disk space.	%DefaultInputFRSDir/temp%
File Store Directory	Specifies the directory where repository objects are stored.	%DefaultInputFRSDir/%
Maximum Retries for File Access	Specifies the number of times the server tries to access a file.	1

Output File Repository Server Settings

PROPERTY	DESCRIPTION	DEFAULT VALUES
Maximum Retries for File Access	Specifies the number of times the server tries to access a file.	1
Maximum Idle Time (minutes)	Specifies the length of time that the server waits before it closes inactive connections. Setting a value that is too low can cause your request to close prematurely. Setting a value that is too high can cause excessive consumption of system resources such as processing time and disk space.	10 minutes

PROPERTY	DESCRIPTION	DEFAULT VALUES
Temporary Directory	Specifies the directory where temporary files are created when necessary.  You may encounter performance issues if this directory does not have adequate disk space.	%DefaultOutputFRSDir/temp%
File Store Directory	Specifies the directory where repository objects are stored.	%DefaultOutputFRSDir/%

INCREASING REPORTING SERVICES

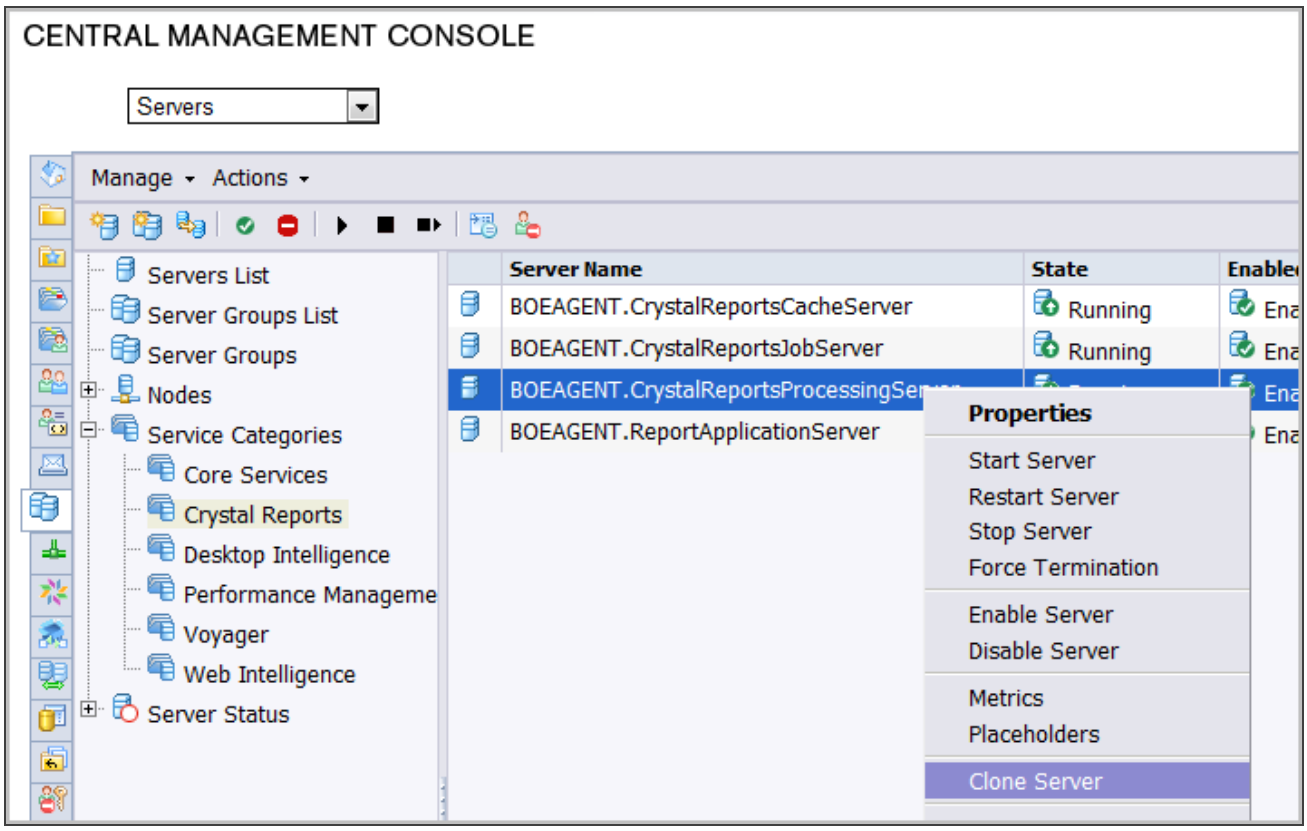
Each service can be replicated on the same server to increase the load capacity. You are limited only by the memory, CPU, and disk capacity of your service. Increasing the services will allow more users to run reports.

An indication that you need to increase the number of services is if your users receive messages from the system informing them that there are insufficient job servers available to process their request.

Add additional services

1. Log on to the CMC using an **Administrator** account.
2. Navigate to the **Servers** list and select the service you would like to duplicate.

3. Right-click on the server name and select **Clone Server**.



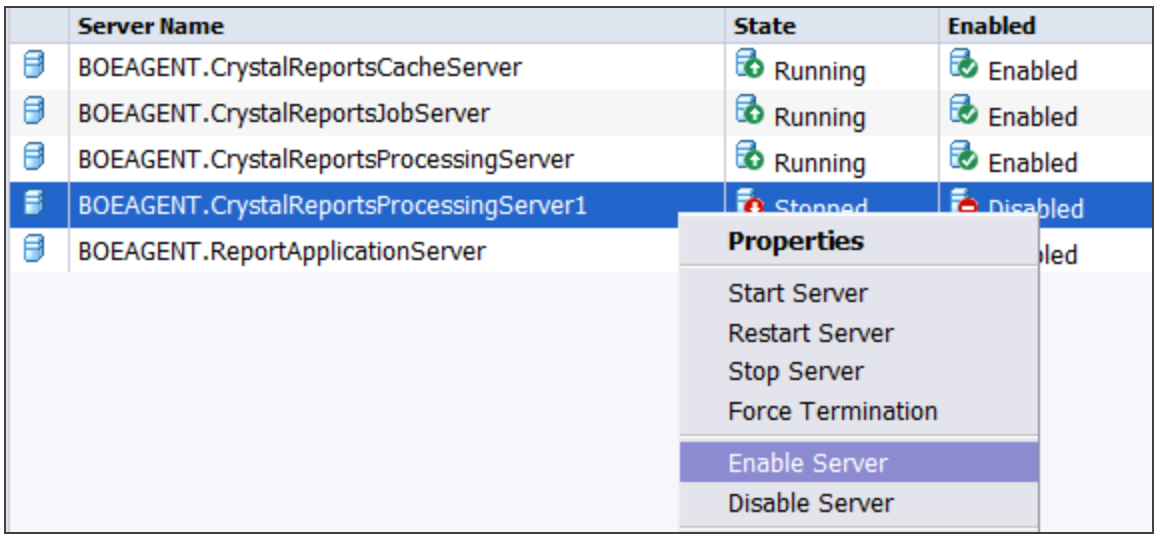
4. You may give this duplicate a new name. By default it is numbered starting at 1.

> If you are in a clustered environment you will need to select the node name.

5. Click **OK** to see the new service in the list.

> By default this service is disabled and not started so that you have time to set the properties before the service starts work. We recommend that the properties of this new service match those of the service you copied.

6. After you are satisfied with the properties, right-click on the service name, and select **Enable Server**.



7. To start the service, right-click on the server name, and select **Start Server**. Your new service is now ready to start processing reports.

You may add additional services at any time, even while users are active in the system.

INCREASING PHYSICAL SERVERS (CLUSTERING)

The report processing services can be replicated on multiple physical servers (Clustering). A cluster consists of two or more servers working together against a common report repository. You should consider adding physical servers if you have many users or the reports you are running are very large, and require significant CPU, memory, and disk storage. A cluster also allows one server to continue processing if another node in the cluster should fail.

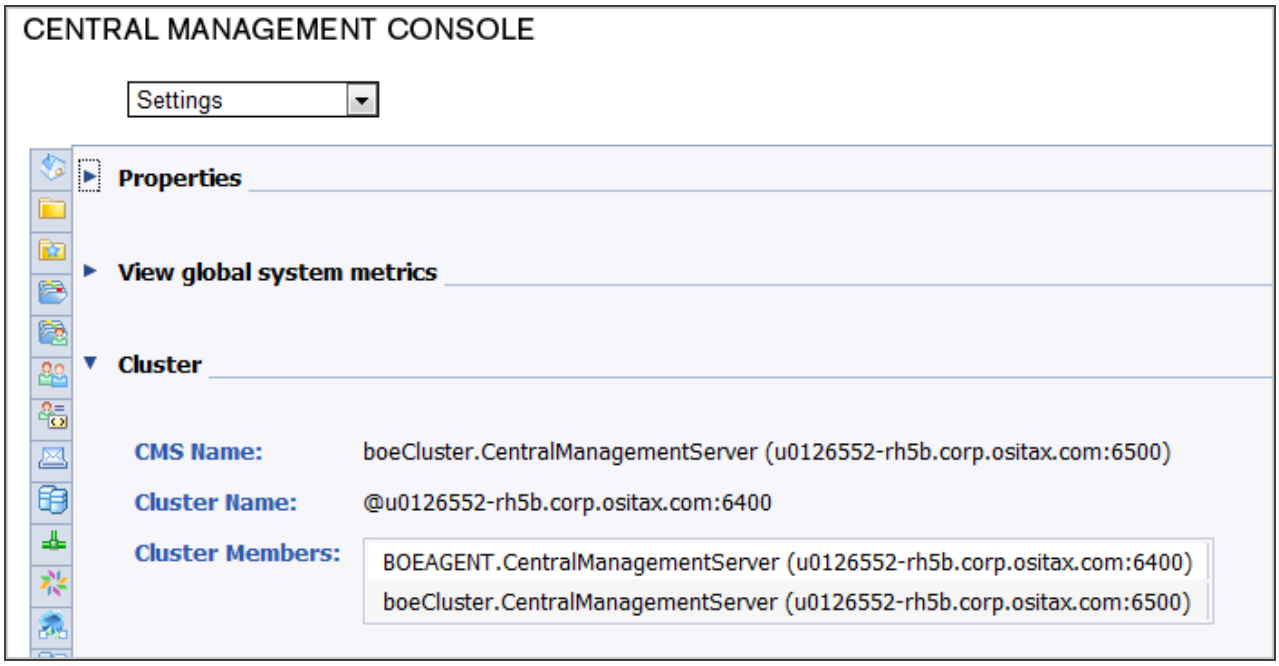
Within a clustered environment, the Input and Output File Repository Servers must not share the same root directory because modifications to the files and subdirectories belonging to one server might damage files belonging to the other.

- If you run multiple File Repository Servers, all Input File Repository Servers must share the same root directory, and all Output File Repository Servers must share the same root directory.
- We recommend that you replicate the root directories using a RAID array or an alternative hardware solution.
- The root directory should be on a drive that is local to the server or on a SAN array.

Adding an additional node to the cluster is simply the process of installing SAP BusinessObjects BI Platform on an additional server and having it share the same Reporting Repository. By selecting the custom installation option and choosing to cluster the installation, each node will be aware of the other. You can see the other nodes in your cluster from the CMC.

View nodes in the cluster

1. Log on to the CMC using an **Administrator** account.
2. Navigate to **Settings** and expand the **Cluster** section.



If you have a multiple servers in your cluster, configure each as you would a single server.

In a clustered configuration you will see the services from all members of the cluster in the CMC.

CONFIGURING THE ETL SERVER

The following sections describe how to configure the ETL server:

ETL CLUSTERING WITH SERVER GROUPS

If you need to ETL a large amount of data and you would like to use more than one physical server to do this, you can cluster the ETL Server by using Server Groups to associate one or more ETL Job Servers on different physical servers. A server group automatically measures resource availability on each ETL Job Server in the group and distributes scheduled batch jobs to the server with the lightest load at runtime.

All of the ETL Job Servers in a server group must be associated with the same ETL repository, which must be defined as a default repository. Each physical server can only contribute one Job Server to a server group.

Each ETL Job Server in the server group must have:

- Identical SAP Data Services versions.
- Identical database server versions.
- Identical database connections (for instance, the TNSNAME name on each server must be identical for Oracle).
- Identical locale.

The requirement that all ETL Job Servers in a server group be associated with the same ETL repository allows you to more easily track which processes are associated with a server group. We recommend that you use a naming convention for server groups that includes the name of the repository. For example, for an ETL repository called DEV, a server group might be called SG_DEV. On start-up, all Job Servers check the repository to find out if they must start as part of a server group.

Compared to stand-alone ETL Job Servers, Job Servers in a server group:

- Collect a list of other Job Servers in their server group.
- Collect system load statistics every 60 seconds.

- Detect the number of CPUs (on start-up only).
- Determine the average CPU load.
- Determine the available virtual memory.
- Process service requests for system load statistics.
- Accept server group execution requests.

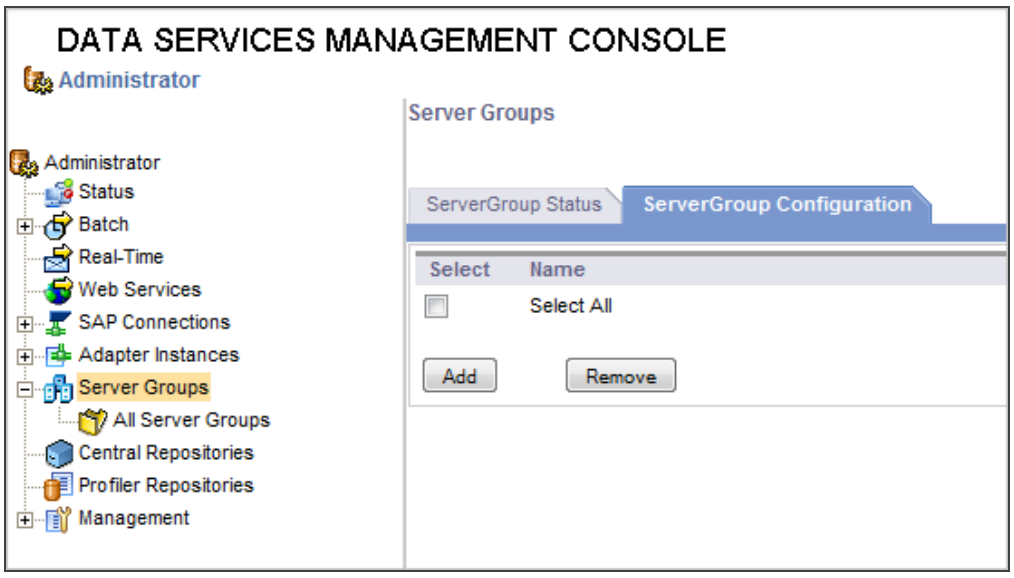
All Job Servers in a server group collect and consolidate system load statistics and convert them into a load balance index value for each Job Server. A Job Server's load balance index value allows the software to normalize statistics taken from different platforms. The Job Server with the lowest index value is selected to execute the current job. The software polls all Job Server computers every 60 seconds to refresh the load balance index.

Create a Server Group

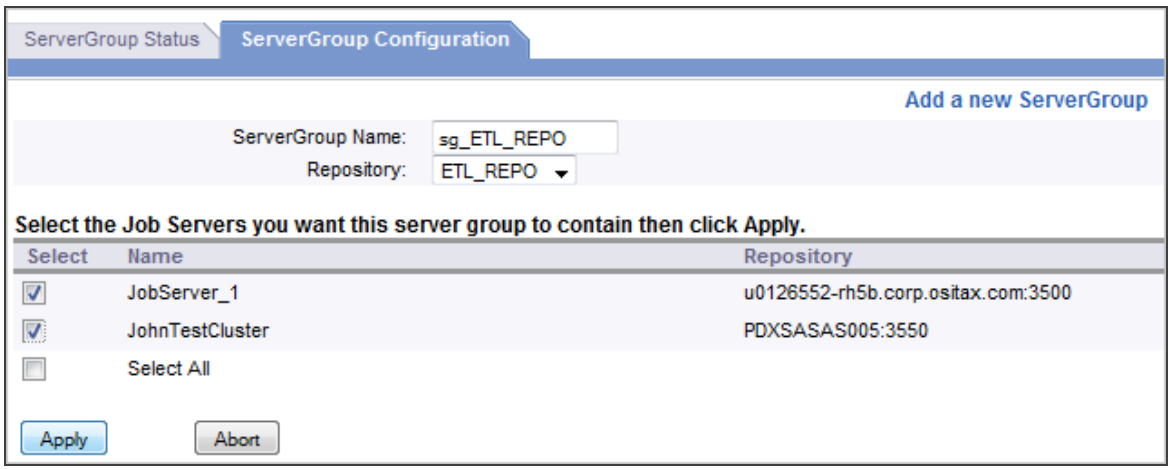
Creating a Server Group requires having more than one ETL Job Server running. Only one ETL Job Server should run per physical server to ensure maximum performance, but you can have any number of ETL Job Servers in a Server Group. See Creating Additional ETL Job Servers (page 38) for details on creating additional ETL Job Servers.

1. Open a browser and enter the Data Services URL (http://<host>:<port>/DataServices).
2. Log on using your **Administrator** account.
3. Navigate to **Server Groups**.

4. Click the **Server Group Configuration** tab and click **Add**.



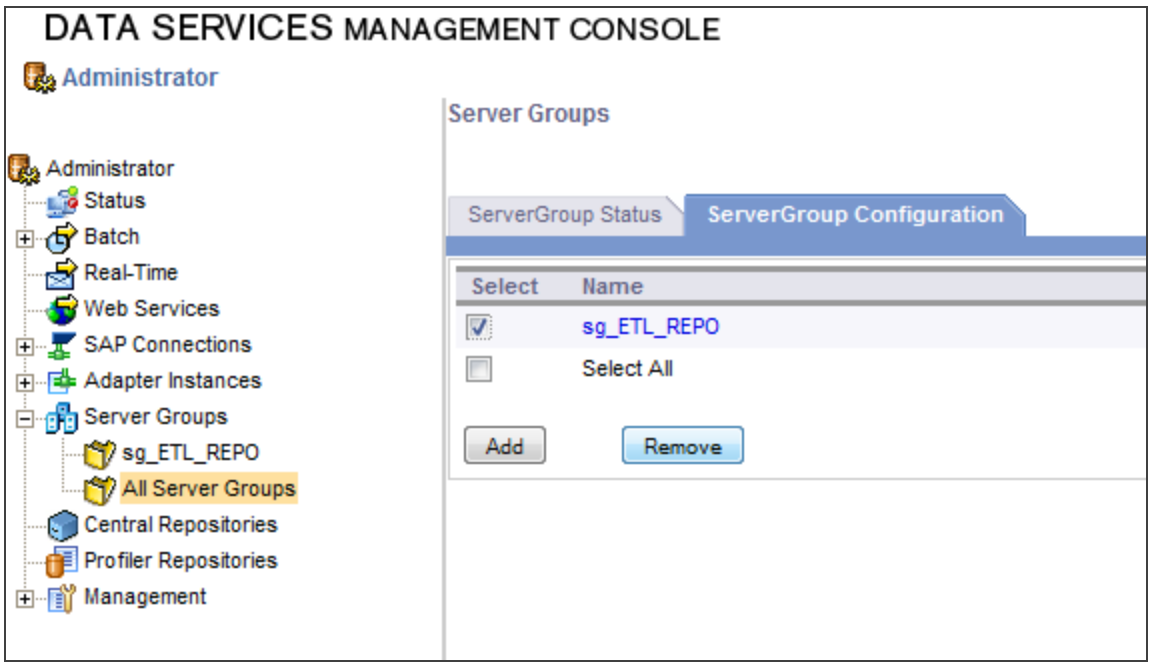
5. Enter the **ServerGroup Name**, select the **Repository**, and set the Job Servers. Then click **Apply**.



Delete a server group

1. Open a browser and enter the Data Services URL (http://<host>:<port>/DataServices).
2. Log on using your **Administrator** account.
3. Navigate to **Server Groups** and select **All Server Groups**.
4. Click the **Server Group Configuration** tab.

5. Select the Server Group you want to delete and click **Remove**.

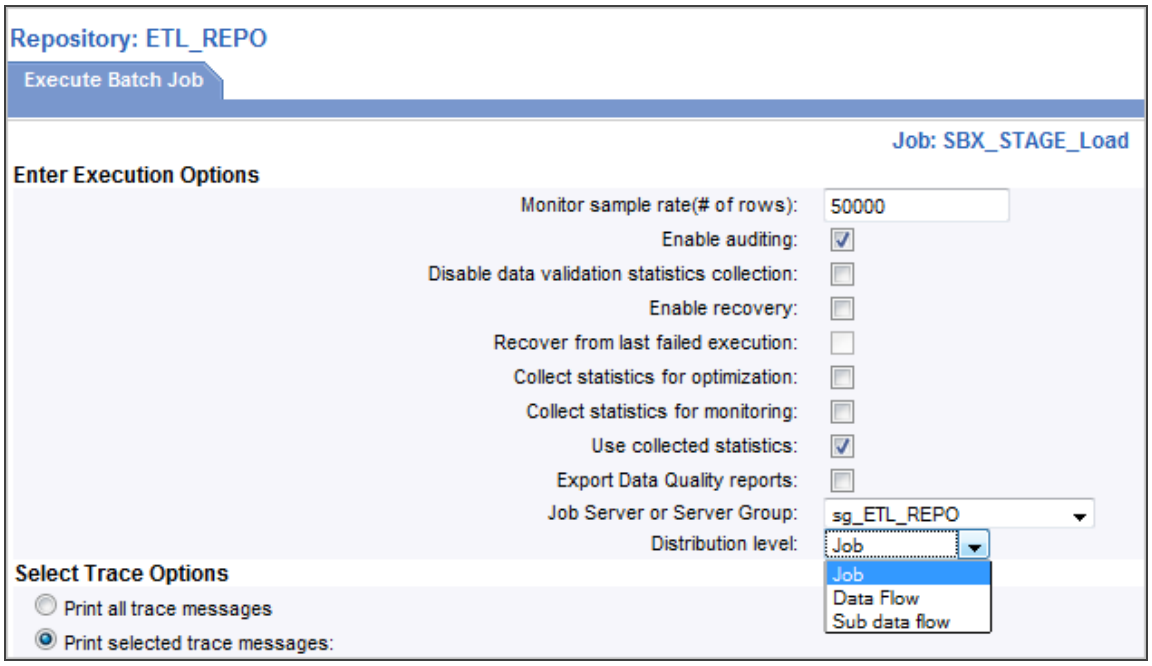


You can execute parts of your job on different Job Servers in a server group. You can select the following distribution levels from the Designer's **Execution Properties** window or from the **Administrator's Execute Batch Job**, **Schedule Batch Job**, and **Export Batch Job** pages:

- **Job level**: An entire ETL process will execute on a single available ETL Job Servers.
- **Data flow level**: Each data flow within a job will execute on one of the available ETL Job Servers.
- **Sub data flow level**: A resource-intensive operation (such as a sort, table comparison, or table lookup) within a data flow can execute on one of the available ETL Job Servers.

Running an ETL Using Server Groups

After you create a server group, you can select a server group to execute a job. Other than having to select a **Server Group** and **Distribution level**, running or scheduling an ETL to a Server Group is the same as running an ETL without a **Server Group**. Simply select your **Server Group** instead of a **Job Server**, and then select the **Distribution level**.



When you execute a job using a server group, the server group uses the Job Server in the group that is running on the computer that has the lightest load. The system Administrator will also need to resynchronize a Job Server with its repository if there are changes made to the server group configuration settings.

You can execute parts of your job on different Job Servers in a server group. You can select the following distribution levels from the Designer's **Execution Properties** window or from the **Administrator's Execute Batch Job**, **Schedule Batch Job**, and **Export Batch Job** pages:

- **Job level**: An entire ETL process will execute on a single available ETL Job Servers.
- **Data flow level**: Each data flow within a job will execute on one of the available ETL Job Servers.
- **Sub data flow level**: A resource-intensive operation (such as a sort, table comparison, or table lookup) within a data flow can execute on one of the available ETL Job Servers.

CREATING ADDITIONAL ETL JOB SERVERS

To distribute the workload of ETL processing, you may want to create additional Job Servers and form a Server Group. See ETL Clustering with Server Groups (page 33) for additional information.

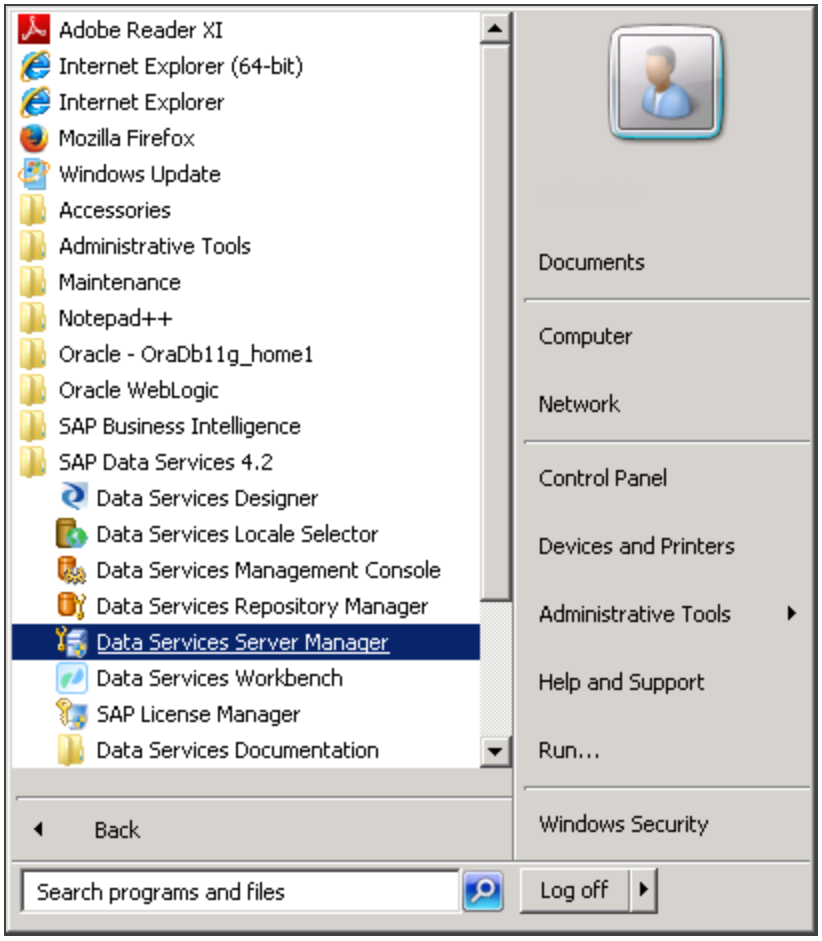
While the system will not prevent you from creating multiple ETL Job Servers on a single physical server, we recommend that you have only one ETL Job Server per machine to maximize performance. This allows the Job Server to make the best use of the available system resources.

Creating Additional ETL Job Servers in Windows

To create additional ETL Job Servers in the Windows Operating system, you must sign into the machine on which you installed SAP BusinessObjects Data Servers.

Create an ETL job server

1. From the**Start** menu, locate **Data Services Server Manager** within the **Data Services** folder.



2. Click **Edit Job Server Config** and then click **Add**.

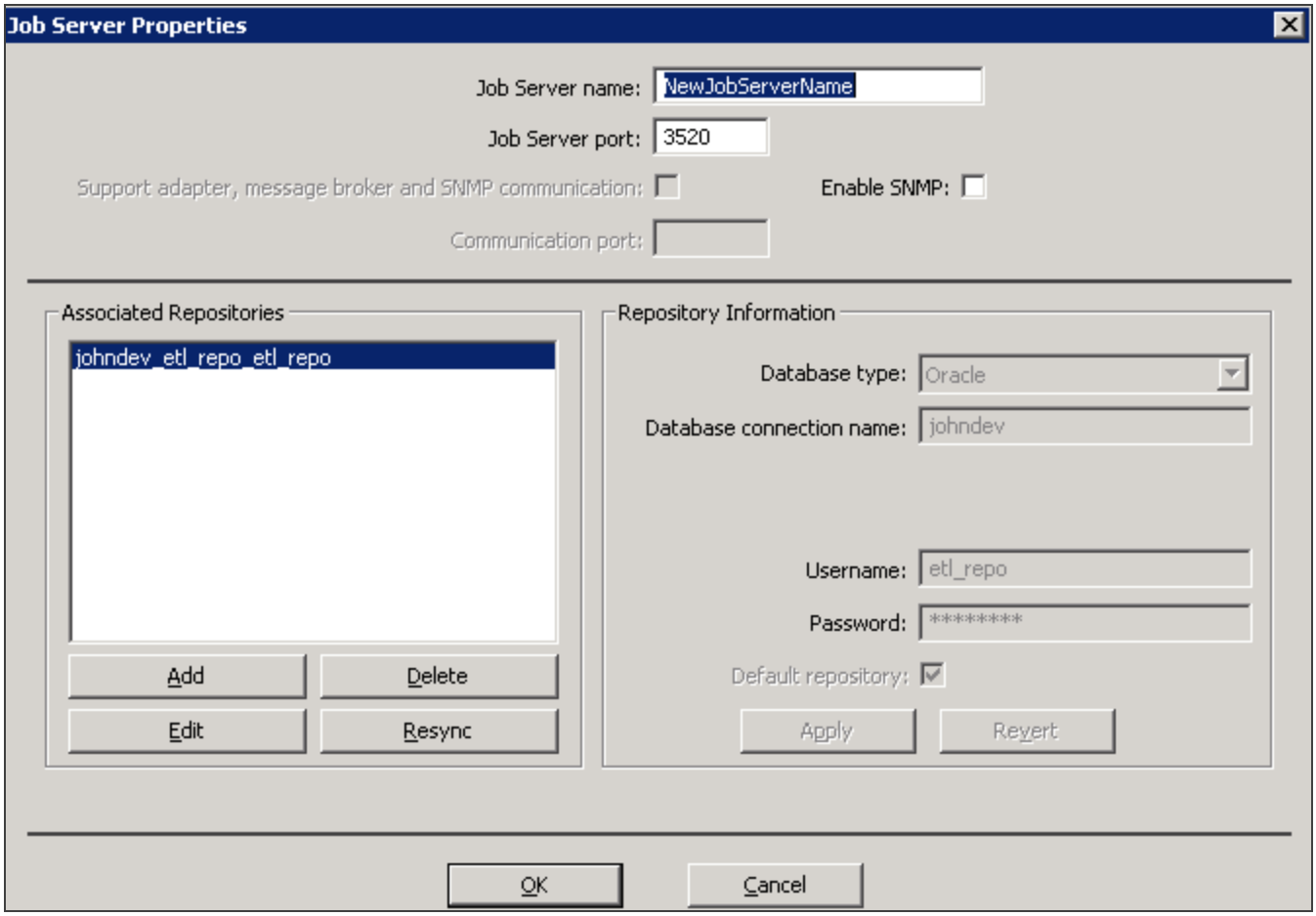
3. Enter the **Job Server** name and enter a unique **Job Server port**. We recommend that you stay within a range such as 3500-3520, for example. If the port you specify is already used, the system notifies you and gives you a chance to pick a different port when you attempt to apply the changes.

4. Click the **Add** button in the **Associated Repositories** section.

5. Select the Database type for the ETL Repository. You will be reusing the same repository you created during the installation process for Reporting.

6. Enter the **Database connection name**. This is the TNS Name entry for Oracle, followed by the ETL Repository schema username (for example, ETL_REPO) and schema password.

7. If desired, you can designate this as the default ETL Job Server. Select the **Default repository** check box.



8. Click **Apply**. The system attempts to connect to the ETL Repository to verify the connection information. Any errors are shown so that you can make corrections and try again.

9. When you are finished, click **OK**. You will now see your new ETL Job Server in the list.

10. Click **OK** to finish. You can then close the Server Manager.

Your new ETL Job Service will automatically be added to the Job Service and started.

Creating Additional ETL Job Servers in Linux

To create additional ETL Job Servers on the Linux operating system, you will need to sign in as the same user who installed Reporting.

Create an ETL job server

1. Navigate to the businessobjects/DataServices/bin directory.
2. Execute the following two commands at the command prompt to start the **Data Services Server Manager Utility**.

```
. ./al_env.sh
svrcfg
```

3. Select **3: Configure Job Server**. You will see a list of the currently defined ETL Job Servers.
4. To create a new Job Server, select **c**.
5. Enter the **Job Server name** for the new ETL Job Server.
6. Enter a unique number for the **Job Server port**. We recommend that you stay within a range such as 3500-3520, for example. If the port you specify is already used, the system notifies you and gives you a chance to pick a different port when you attempt to apply the changes.

You may receive a message telling you that another Job Server is managing the adapters. This is expected as only one ETL Job Server can be set to manage the adapters.

7. You will be prompted to enable SNMP for this new Job Server. Enter **N** (No).
8. Select the database type for the ETL Repository. You will be reusing the same repository you created during the installation process for Reporting.
9. Enter the connection information. This is the TNSNames entry for Oracle, followed by the ETL Repository schema name (for example, ETL_REPO) and the schema password.

10. You will be asked to confirm the information. If it is correct, enter **Y** (Yes); otherwise, enter **N** (No) and you will be able to make corrections.

    You will see messages similar to the following:

    Updating the repository <ETL_REPO@sabrixdb__Oracle>. Please wait...

    SNMP has been enabled for this machine, but not for any JobServers on it. You must enable SNMP on one or more JobServers.

11. Next, you will be prompted to continue modifying Job Servers. Enter **N** (No) to return to the main menu from which you can exit.

Your new ETL Job Service will automatically be added to the Job Service and started.

ETL EMAIL NOTIFICATIONS

The Job Server allows for each ETL run to send automatic email notifications for a success or failure. This requires the use of Simple Mail Transfer Protocol (SMTP). In order to setup the Job Server for email notification capabilities perform the following steps.

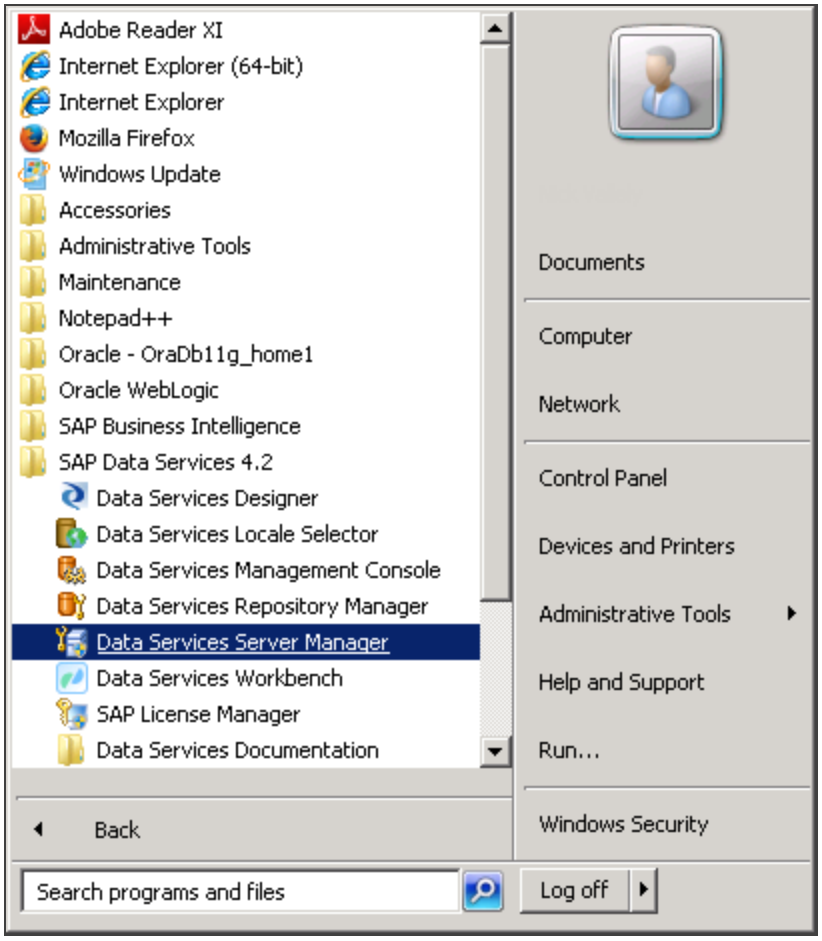
1. Add your SMTP information to the system.
2. Provide the address of your SMTP server—either an IP Address or the fully qualified domain name of the server.
3. Define the email address to use as the **From** in email messages. (This does not have to be a monitored email account.)

Setting up Email Notifications in Windows

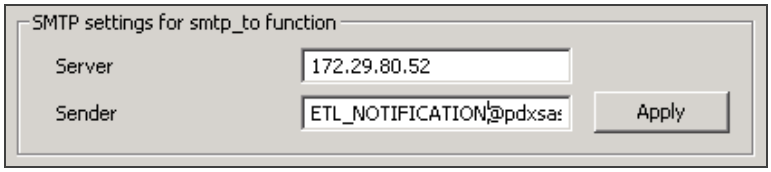
To add email notifications to your ETL Job Server on the Windows Operating system, you need to sign into the machine on which you installed SAP Data Services.

Create email notifications

1. From the **Start** menu, locate **Data Services Server Manager** within the **Data Services** folder.



2. Enter the name or IP Address of your SMTP server in the SMTP settings.
3. Enter the name of the sender of the notifications. For example, ETL_Notifications@yourcompany.com. This account must include the @ symbol.
4. Click **Apply** to save your changes.



5. You must restart the ETL Job Server for this change to take effect. To do so, click **Restart**. Once you have restarted, you can close the Server Manager.

Setting up Email Notifications in Linux

To set up email notifications on the Linux operating system, sign in as the same user who installed Reporting. After you log in:

1. Navigate to the *businessobjects/DataServices/bin* directory.
2. Execute the following 2 commands at the command prompt to start the **Data Services Server Manager Utility**:

```
. ./al_env.sh
svrcfg
```

3. Select **7: Configure SMTP**. Then enter **e** to edit the existing configuration.
4. Enter the name or IP address of your SMTP server.
5. Enter the name of the sender of the notifications. For example, ETL_Notifications@yourcompany.com. This account must include the @ symbol. You will then see your new settings.
6. Press **e** if you need to make any corrections.
7. Press **q** to quit and return to the main menu.

ETL LOG RETENTION

When each ETL process is run, a set of logs is created. You may want to adjust the retention period of these logs.

Adjust ETL log retention

1. Log in to the Data Services Management Console using an administrative account.

2. Navigate to **Management**.

3. Select the **Log Retention Period** to change the retention periods. By default system logs will be retained 60 days and ETL Job Server logs will be retained for 15 days.

4. After making your choices, click **Apply**.