Monitoring SAP Business Objects

eG Enterprise v6.0

Restricted Rights Legend

The information contained in this document is confidential and subject to change without notice. No part of this document may be reproduced or disclosed to others without the prior permission of eG Innovations Inc. eG Innovations Inc. makes no warranty of any kind with regard to the software and documentation, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose.

Trademarks

Microsoft Windows, Windows NT, Windows 2003, Windows 2003 and Windows 2008 are either registered trademarks or trademarks of Microsoft Corporation in United States and/or other countries.

The names of actual companies and products mentioned herein may be the trademarks of their respective owners.

Copyright

Table of Contents

Table of Figures

1

Monitoring SAP Business Objects

SAP BusinessObjects BI (also known as BO or BOBJ) is a suite of front-end applications that allow business users to view, sort and analyze business intelligence data.

The suite includes the following key applications:

- Crystal Reports -- Enables users to design and generate reports
- Xcelsius/Dashboards -- Allows users to create interactive dashboards that contain charts and graphs for visualizing data
- Web Intelligence -- Provides a self-service environment for creating ad hoc queries and analysis of data both online and offline
- Explorer -- Allows users to search through BI data sources using an iTunes-like interface. Users do not have to create queries to search the data and results are shown with a chart that indicates the best information match.

SAP BusinessObjects Business Intelligence platform can be thought of as a series of conceptual tiers:

- **Client tier:** The client tier contains all desktop client applications that interact with the SAP BusinessObjects Business Intelligence platform to provide a variety of reporting, analytic, and administrative capabilities. Examples include the Central Configuration Manager (BI platform installation program), Information design tool (BI platform Client Tools installation program), and SAP Crystal Reports 2011 (available and installed separately).
- **Web tier:** The web tier contains web applications deployed to a Java web application server. Web applications provide SAP BusinessObjects Business Intelligence platform functionality to end users through a web browser. Examples of web applications include the Central Management Console (CMC) administrative web interface and BI launch pad. The web tier also contains Web Services. Web Services provides SAP BusinessObjects Business Intelligence platform functionality to software tools via the web application server, such as session authentication, user privilege management, scheduling, search, administration, reporting, and query management. For example, Live Office is a product that uses Web Services to integrate SAP BusinessObjects Business Intelligence platform reporting into Microsoft Office products.
- **Management tier:** The management tier (also known as intelligence tier) coordinates and controls all of the components that make up SAP BusinessObjects Business Intelligence platform. It is comprised of the Central Management Server (CMS) and the Event Server and associated services. The CMS provides maintains security and configuration information, sends service requests to servers, manages auditing, and maintains the CMS system database. The Event Server manages file based events, which occur in the storage tier.
- **Storage tier:** The storage tier is responsible to handling files, such as documents and reports. The Input File

Repository Server manages files that contain information to be used in reports, such as the following file types: .rpt, .car, .exe, .bat, .js, .xls, .doc, .ppt, .rtf, .txt, .pdf, .wid, .rep, .unv. The Output File Repository Server manages reports created by the system, such as the following file types: .rpt, .csv, .xls, .doc, .rtf, .txt, .pdf, .wid, .rep. The storage tier also handles report caching to save system resources when users access reports.

- **Processing tier:** The processing tier analyzes data and produces reports. This is the only tier that accesses the databases that contain report data. This tier is comprised of the Adaptive Job Server, Connection Server (32- and 64-bit), and processing servers such as the Adaptive Processing Server or Crystal Reports Processing Server.
- **Data tier:** The data tier consists of the database servers hosting the CMS system database and Auditing Data tore. It also consists of any database servers containing relational, OLAP, or other data types for reporting and analytic applications.

SAP BusinessObjects Business Intelligence platform consists of collections of servers running on one or more hosts. Small installations (such as test or development systems) can use a single host for a web application server, database server, and all BI platform servers. Medium and large installations can have servers running on multiple hosts. Large installations can have several BI platform server hosts working together in a cluster. The term server is used to describe an operating system level process (on some systems, this is referred to as a daemon) hosting one or more services. For example, the Central Management Server (CMS) and Adaptive Processing Server are servers. A server runs under a specific operating system account and has its own PID. A service is a server subsystem that performs a specific function. The service runs within the memory space of its server under the process ID of the parent container (server). For example, the Web Intelligence Scheduling Service is a subsystem that runs on the Adaptive Job Server. A node is a collection of BI platform servers running on the same host and managed by the same Server Intelligence Agent (SIA). One or more nodes can be on a single host.

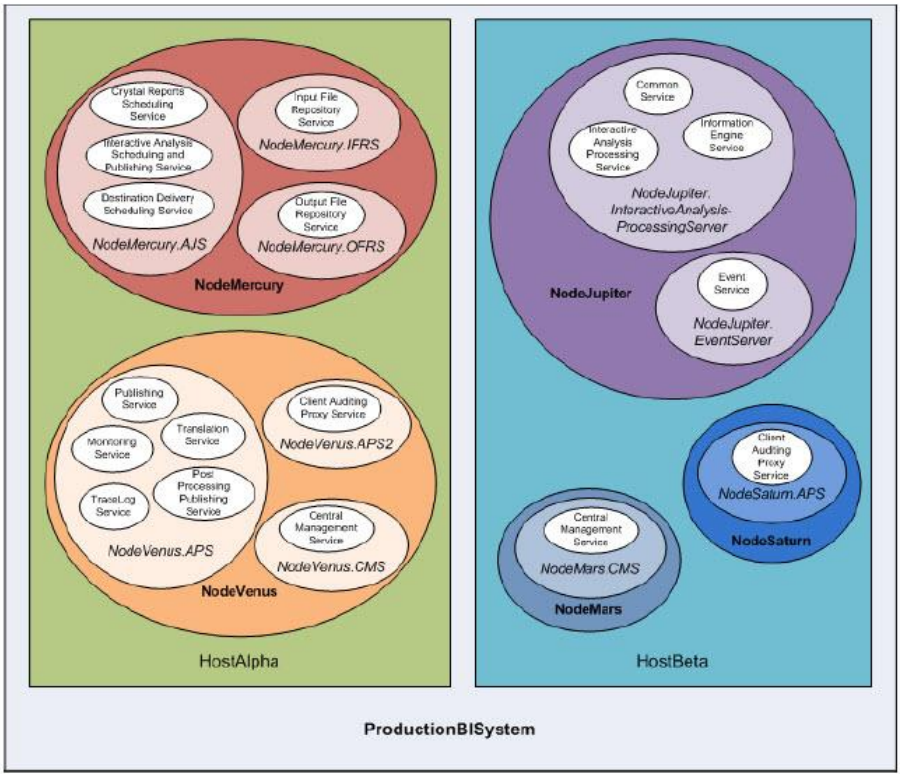

Figure 1. 1: Architecture of SAP BusinessObjects BusinessIntelligence Platform

If a single server/service on a node fails or processes requests slowly, then the user experience with the

corresponding front-end application will suffer, making analysis of business intelligence data difficult and delaying crucial business decisions. If this is to be avoided, then every server/service running in a node should be closely monitored and administrators should be proactively notified of performance setbacks that the servers/services experience. eG Enterprise provides a specialized, web-based *SAP BOBI* monitoring model for this purpose.

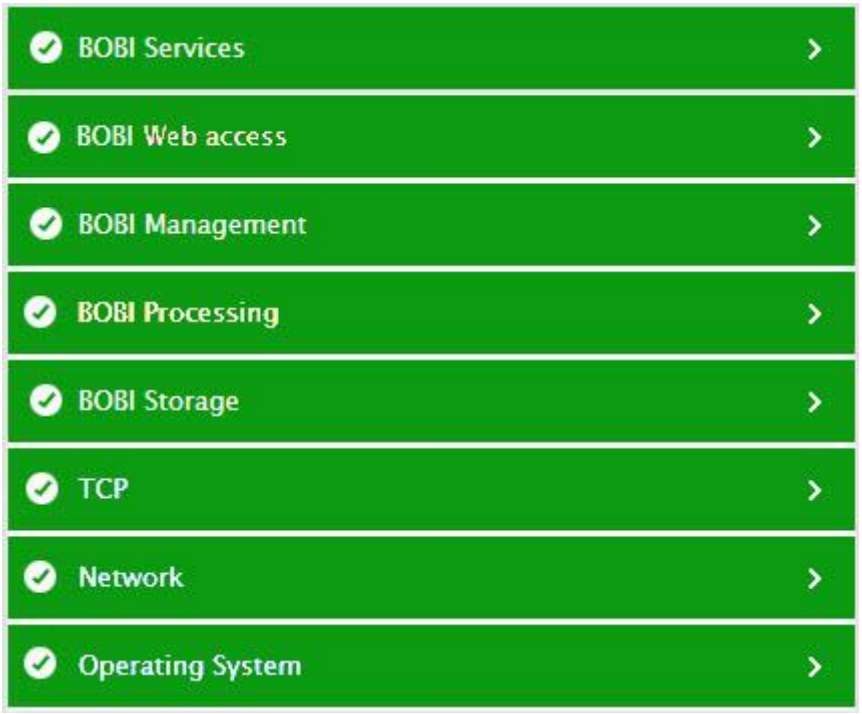

Figure 1. 2: The layer model of the SAP BOBI platform

Each layer of Figure 1.2 above is mapped to tests that monitor the overall health, availability, and performance of each of the servers running in a configured node. Using the metrics so reported, administrators can find quick and accurate answers for the following performance queries:

- Are the various BOBI services in healthy state? Is the node healthy ?
- What is the system load in terms of publications, instances running, reports scheduled etc.,?
- What are the errors occurring in the various Crystal Reports servers defined?
- What is the load generated by the Multi Dimensional Analysis service in terms of queries running, sessions and cubes created?
- How many resource intensive queries are running for the Data Federation service? What is the disk and memory utilization of queries run by this service? How many queries have failed? What is the current number of connections created?
- Are there any Dashboards server errors?
- How is the Platform Search Service performing?
- Are there any errors from connection servers, web intelligence servers etc.,?

- Is the BI launchpad available? What is the response time like?
- Is the Central Management Server auditing properly? How many users are currently using the server? What is their average response time?
- Have any Central management server jobs failed? How many jobs are pending?
- Are the Job Server destinations available? Are there are errors in the server?
- What is the performance of the Adaptive Processing Server JVM? Are there any errors in this server?
- What is the utilization and response time of the Dashboards processing server?
- How many connections are used by the Data Semantic Layer Bridge? How many queries are being run for this service?
- How many documents is the Report Application Server processing? What is its thread utilization?
- How many CORBA requests is the web intelligence server processing? How many users are using the web intelligence server? Are too many sessions getting timed-out?
- How is the web intelligence server cache utilization? What is the cache swap rate?
- Is the web intelligence server using too much memory ? What is its CPU utilization ?
- How is the Dashboards cache server performing? Is it slow?
- How many files are accessed through the file repository server? How many connections? What are the data transfer rates?
- How is the health of each configured server? Which servers are enabled and are they running properly?
- What is the rate of error messages occurring in the various servers running in this node? What are those error messages?
- Is the root directory of the file repository server's partition running out of free space?
- Are the Bobi server processes running? How much CPU and memory are they using the node's computer?
- What is the performance of the host computer on which the node has been installed? How is its connectivity?

1.1 How does eG Enterprise Monitor the SAP BOBI Platform?

To collect all the metrics summarized above, it is recommended that you use eG's Agent-based Monitor. This Monitor should then be configured to connect to the SAP BOBI platform via JMX and obtain the status and performance information of the node configured for monitoring, using the attributes exposed by BOBI's Managed beans.

To enable the Monitor to establish this connection and collect metrics, the following pre-requisites should be fulfilled:

- The **Monitoring Application** of SAP BOBI should be up and running for the eG Monitor to perform metrics collection. The Monitoring application allows you to capture the runtime and historical metrics of BI platform servers, for reporting and notification. To know how to enable the **Monitoring Application**, refer to Section 1.1.1 below.

- When managing the *SAP BOBI* component in the eG administrative interface, make sure you manage it using the port number of the web application server on which SAP BOBI runs. For instance, if the default tomcat application server is used for the SAP BOBI installation and its default port number has not been changed, then specify 8080 as the **Port number** of the *SAP BOBI* component when managing it.

- Each test run by the eG Monitor on the *SAP BOBI* should be configured with the following:

    - The **RMI PORT** number of SAP BOBI's Monitoring Application. To know how to determine the RMI port of the Monitoring Application, refer to Section 1.1.1 below.

    - The **JNDI NAME** of SAP BOBI's Monitoring Application. To know how to determine the JNDI name of the Monitoring Application, refer to Section 1.1.1 below.

    - A **JMX USER** and **JMX PASSWORD**. Here, provide the credentials of a JMX user who fulfills the following conditions:

        - The **Authentication Type** of the user should be **Enterprise**.

        - The user should have access rights to the BOBI Monitoring Application.

        - The user should belong to the default *Monitoring users* group on SAP BOBI.

1.1.1 Enabling the Monitoring Application of the SAP BOBI Platform

To check the status of the Monitoring Application, enable it (if required, and determine its JNDI Name and RMI Port, do the following:

1. Access the **BOBI Central Management Console** using a browser. When Figure 1.3 appears, login as a user who belongs to the *administrators* group.

Figure 1. 3: Logging into the Central Management Console

2. The CMC Home Page then appears (see Figure 1.4). Click the **Applications** link indicated in Figure 1.4.

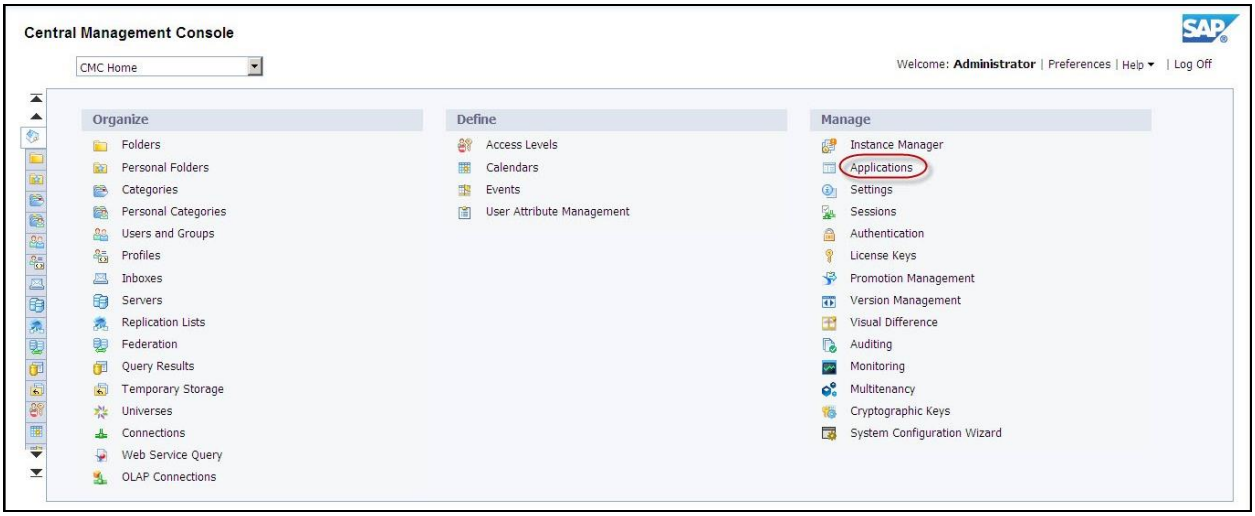

Figure 1. 4: Clicking the Applications link in the CMC Home Page

3. Then, select the **Monitoring Application** from the **Applications** list of Figure 1.5. To access the properties of the chosen application, select the **Properties** option from the **Manage** menu, as shown by Figure 1.5.

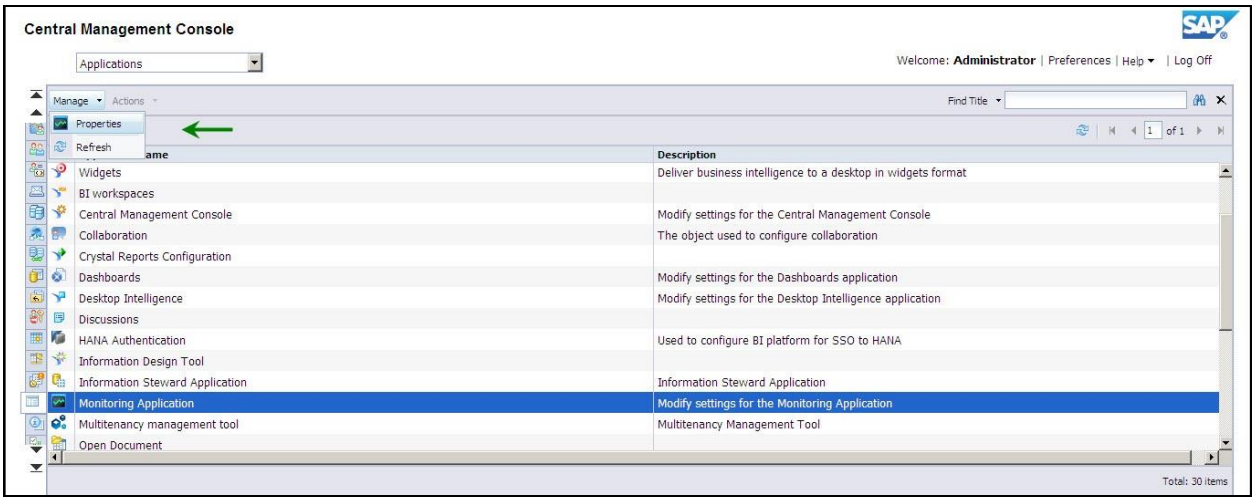

Figure 1. 5: Accessing the properties of the Monitoring Application

4. The **Monitoring Application Properties** page (see Figure 1.6) will then appear. Here, select the **Enable Monitoring Application** check box to enable the Monitoring Application.

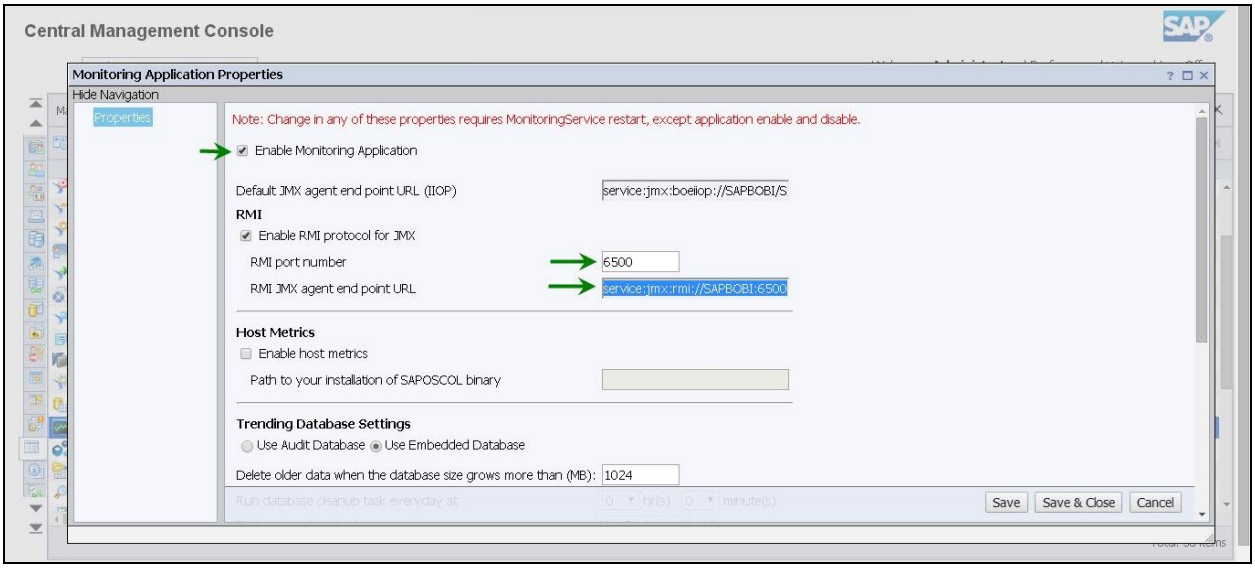

Figure 1. 6: The Properties of the Monitoring Application

5. Also, select the **Enable RMI protocol for JMX** check box in Figure 1.6 to activate the RMI Port. Use the **RMI port number** displayed in Figure 1.6 to configure the **JMX REMOTE PORT** parameter of the eG tests.

6. Likewise, you can also obtain the value for the **JNDI NAME** parameter from Figure 1.6. For this, first focus on the **RMI JMX agent end point URL** specification in Figure 1.6. The JNDI Name is part of this URL. Figure 1.7 below displays a sample URL and indicates which part of it forms the JNDI name.

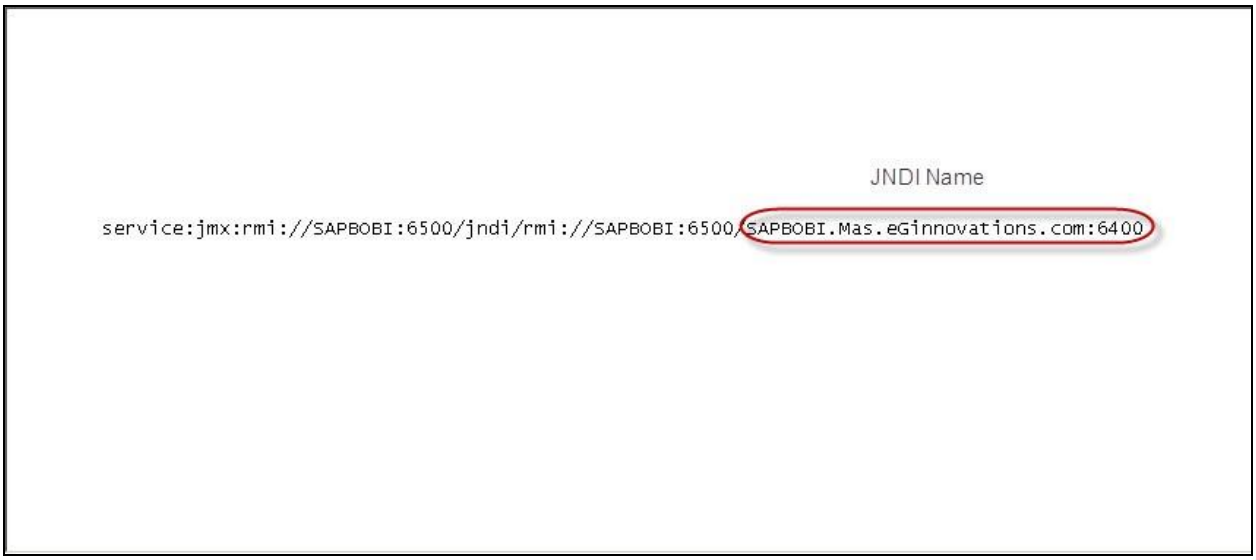

Figure 1. 7: Determining the JNDI name from the RMI JMX agent end point URL

7. Finally, click the **Save & Close** button in Figure 1.6 to register the changes made to the properties of the Monitoring Application.

The sections that follow discuss each of the top 5 layers of of Figure 1.2 in detail. The remaining layers have already been discussed in the *Monitoring Unix and Windows Servers* document.

1.2 The BOBI Storage Layer

This layer reports the status and performance of the Dashboards Cache and File Repository servers.

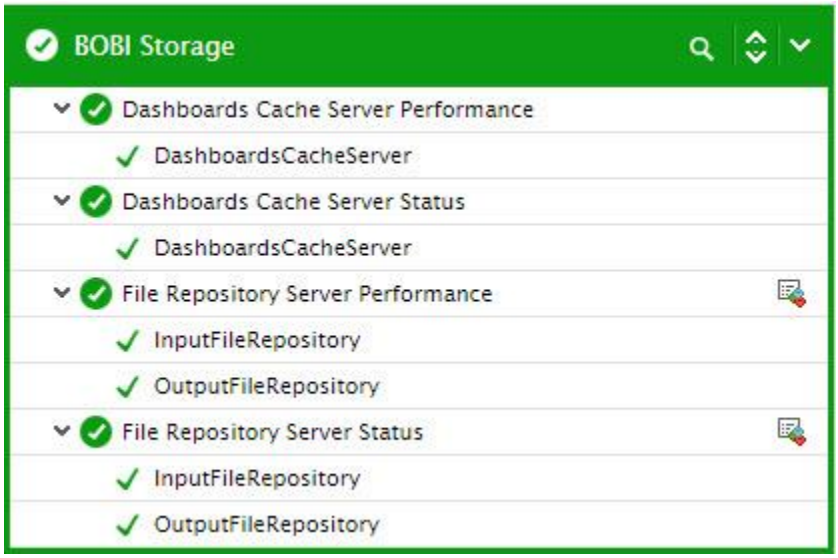

Figure 1. 8: The tests mapped to the BOBI Storage layer

1.2.1 Dashboards Cache Server Status Test

The Dashboards Cache Server intercepts report requests sent from clients to the Dashboard server. If the cache server cannot fulfill the request with a cached report page, it passes the request on to the Dashboard server, which runs the report and returns the results. The cache server then caches the report page for potential future use.

If the cache server is down, then all report requests will be sent directly to the Dashboard server for servicing, resulting in additional processing overheads. If the cache server is slow, report requests will be processed slowly, increasing user dissatisfaction with the service. By promptly capturing the failure/slowness of a Dashboards Cache server and rapidly taking action against such anomalies, administrators can ensure the continuous availability and peak performance of the cache server. The **Dashboards Cache Server Status** test aids administrators in this endeavor! This test tracks the health, status, and thread pool usage of the Dashboards cache server and alerts administrators to probable deviations in the availability and performance of the server. This way, the test allows administrators adequate time to take pre-emptive action against the issues noticed, so that guaranteed cache server performance levels are maintained at all times.

Purpose	Tracks the health, status, and thread pool usage of the Dashboards cache server and alerts administrators to probable deviations in the availability and performance of the server
Target of the test	A SAP BOBI node
Agent deploying the test	An internal/remote agent

Configurable parameters for the test	1. **TEST PERIOD** - How often should the test be executed 2. **HOST** - Host name of the server for which the test is to be configured 3. **PORT** - Enter the port to which the specified **HOST** listens. This should be the port at which the web application server hosting SAP BOBI listens. 4. **JMX REMOTE PORT** - Specify the RMI port number of the BOBI monitoring application. To know the RMI port number of the monitoring application, refer to Section 1.1.1 of this document. 5. **JNDI NAME** - Specify the lookup name for connecting to the JMX connector of the BOBI monitoring application. To know the JNDI name, refer to Section 1.1.1 of this document. 6. **JMX USER** and **JMX PASSWORD** – Enter the credentials of an enterprise authenticated BOBI user belonging to the default *monitoring users* group. 7. **CONFIRM PASSWORD** - Confirm the password by retyping it here. 8. **NODE NAME** – Specify the name of the BOBI node being monitored. 9. **DETAILED DIAGNOSIS** - To make diagnosis more efficient and accurate, the eG Enterprise suite embeds an optional detailed diagnostic capability. With this capability, the eG agents can be configured to run detailed, more elaborate tests as and when specific problems are detected. To enable the detailed diagnosis capability of this test for a particular server, choose the On option. To disable the capability, click on the Off option. The option to selectively enable/disable the detailed diagnosis capability will be available only if the following conditions are fulfilled: • The eG manager license should allow the detailed diagnosis capability • Both the normal and abnormal frequencies configured for the detailed diagnosis measures should not be 0.		
Outputs of the test	One set of results for the Dashboards cache server running in the node monitored		
Measurements made by the	**Measurement**	**Measurement Unit**	**Interpretation**

<table>
<tr>
<td>test</td>
<td>Health state:

Indicates the current health state of the Dashboards Cache server in the monitored node.</td>
<td></td>
<td>The values that this measure can report and their corresponding numeric values are discussed in the table below:

<table><tr><th>Measure Value</th><th>Numeric Value</th></tr><tr><td>Danger</td><td>0</td></tr><tr><td>Caution</td><td>1</td></tr><tr><td>Healthy</td><td>2</td></tr></table>
If the server is not healthy, then the detailed diagnosis of this measure, if enabled, provides the process ID and processing plugin name of the server.

Note:

By default, this measure reports the Measure Values listed in the table above to indicate the health state of the server. In the graph of this measure however, the same is represented using the numeric values only.</td>
</tr>
<tr>
<td></td>
<td>Server running state:

Indicates the current running state of the Dashboards Cache server.</td>
<td></td>
<td>The values that this measure can report and their corresponding numeric values are discussed in the table below:

<table><tr><th>Measure Value</th><th>Numeric Value</th></tr><tr><td>STOPPED</td><td>0</td></tr><tr><td>STARTING</td><td>1</td></tr><tr><td>INITIALIZING</td><td>2</td></tr><tr><td>RUNNING</td><td>3</td></tr><tr><td>STOPPING</td><td>4</td></tr><tr><td>FAILED</td><td>5</td></tr><tr><td>RUNNING_WITH_ERRORS</td><td>6</td></tr><tr><td>RUNNING_WITH_WARNINGS</td><td>7</td></tr></table>
Note:

By default, this measure reports the Measure Values listed in the table above to indicate the running state of the server. In the graph of this measure however, the same is represented using the numeric values only.</td>
</tr>
</table>

	Server enabled state: Indicates whether/not the Dashboards Cache server is enabled.		The values that this measure can report and their corresponding numeric values are discussed in the table below: **Measure Value** / **Numeric Value** Disabled / 0 Enabled / 1 **Note:** By default, this measure reports the **Measure Value**s listed in the table above to indicate whether/not the server is enabled. In the graph of this measure however, the same is represented using the numeric values only.
	Busy server threads: Indicates the number of server threads that are currently servicing requests.	Number	If this measure reaches the configured maximum thread pool size for the server, new requests to the server would have to wait until a server thread becomes free. If this happens often, it may significantly slowdown request processing by the server. In such a situation, you may want to consider resizing the thread pool.

1.2.2 Dashboards Cache Server Performance Test

Ideally, the cache server should be able to serve maximum report requests. If it cannot, then client queries will be routed to the Dashboard Processing Server, which will fulfill the requests by running the queries on the database. This will considerably increase databases accesses and related processing overheads. To minimize these overheads, administrators should continuously monitor report requests, measure how quickly and efficiently the cache server services these requests, and detect deficiencies in cache usage/performance (if any). This is exactly what the **Dashboards Cache Server Performance** test does. This test reports how well the Database Cache Server is utilized and how quickly it processes requests. In the process, the test points to poor cache usage and processing bottlenecks, and also reveals if it is because the server is badly sized.

Purpose	Reports how well the Database Cache Server is utilized and how quickly it processes requests. In the process, the test points to poor cache usage and processing bottlenecks, and also reveals if it is because the server is badly sized.
Target of the test	A SAP BOBI node
Agent deploying the test	An internal/remote agent

Configurable parameters for the test	1. **TEST PERIOD** - How often should the test be executed 2. **HOST** - Host name of the server for which the test is to be configured 3. **PORT** - Enter the port to which the specified **HOST** listens. This should be the port at which the web application server hosting SAP BOBI listens. 4. **JMX REMOTE PORT** - Specify the RMI port number of the BOBI monitoring application. To know the RMI port number of the monitoring application, refer to Section 1.1.1 of this document. 5. **JNDI NAME** - Specify the lookup name for connecting to the JMX connector of the BOBI monitoring application. To know the JNDI name, refer to Section 1.1.1 of this document. 6. **JMX USER** and **JMX PASSWORD** – Enter the credentials of an enterprise authenticated BOBI user belonging to the default *monitoring users* group. 7. **CONFIRM PASSWORD** - Confirm the password by retyping it here. 8. **NODE NAME** – Specify the name of the BOBI node being monitored.		
Outputs of the test	One set of results for the Database Cache Server in the node monitored		
Measurements made by the test	**Measurement**	**Measurement Unit**	**Interpretation**
	Cache hit rate: Indicates the percentage of requests, over the last 500 requests, that have been served with cached data.	Percent	A value over 80% is ideal. A value less than 50% is a cause for concern, as it indicates that a vast majority of requests for reports is not served by the cache server. This increases database accesses and related overheads, and hence, should be avoided.
	Requests served rate: Indicates the rate at which the server serviced requests.	Requests/Sec	A low value or a steady drop in the value of this measure is indicative of a processing bottleneck.
	Average processing time: Indicates the average time taken by the server to process a request.	Millisecs	A low value is desired for this measure. A high value indicates that the server is taking too much time to process requests. This could be owing to a processing bottleneck and warrants closer scrutiny.
	Queued requests: Indicates the number of requests that are either waiting for processing or are being processed.	Number	A steady rise in the value of this measure is a sign of a request queue that is growing. This means that the server is unable to process page requests as quickly as it receives them.
	Current cache size: Indicates the current size of the cache.	MB	A large cache size may be necessary if the server needs to handle large numbers of queries, or highly complex queries. You may want to increase the cache size if the *Cache hit rate, Requests served rate*, and *Average processing time* measures exhibit disturbing trends.

	Open connections: Indicates the number of connections open between the server and its clients.	Number	
	Data transfer rate: Indicates the rate at which data is transferred from the server to its clients.	KB/Sec	
	Current number of auditing events queued: Indicates the number of auditing events that this server has recorded, but which have not yet been retrieved by the CMS Auditor.	Number	If this number increases without bound, it could mean indicate that auditing has not been configured properly or that the system is heavily loaded and generating auditing events faster than the auditor can retrieve them. When stopping servers, it is advisable to disable them first and wait for auditing events to be fully retrieved and this queue becomes empty. Otherwise, they may be retrieved only when this server has been restarted and the CMS polls for them.

1.2.3 File Repository Server Status Test

The File Repository server is responsible for the creation of file system objects, such as exported reports, and imported files in non-native formats. An Input FRS stores report and program objects that have been published to the system by administrators or end users. An Output FRS stores all of the report instances generated by the Job Server. In the absence of the File Repository server, these file system objects cannot be stored, resulting in significant loss of reports and program objects. If this loss is to be prevented, then administrators will have to make sure that the File Repository server is always available and is processing requests speedily. To ascertain this, administrators can use the **File Repository Server Status** test. At configured intervals, this test verifies the overall health, current state, and thread pool usage of the File Repository server, and captures potential performance aberrations well before they occur. This way, the test red flags future anomalies and enables administrators to prevent them from occurring.

Purpose	Verifies the overall health, current state, and thread pool usage of the File Repository server, and captures potential performance aberrations well before they occur. This way, the test red flags future anomalies and enables administrators to prevent them from occurring.
Target of the test	A SAP BOBI node
Agent deploying the test	An internal/remote agent

<table>
<tr><td>Configurable parameters for the test</td><td colspan="3">1. TEST PERIOD - How often should the test be executed
2. HOST - Host name of the server for which the test is to be configured
3. PORT - Enter the port to which the specified HOST listens. This should be the port at which the web application server hosting SAP BOBI listens.
4. JMX REMOTE PORT - Specify the RMI port number of the BOBI monitoring application. To know the RMI port number of the monitoring application, refer to Section 1.1.1 of this document.
5. JNDI NAME - Specify the lookup name for connecting to the JMX connector of the BOBI monitoring application. To know the JNDI name, refer to Section 1.1.1 of this document.
6. JMX USER and JMX PASSWORD – Enter the credentials of an enterprise authenticated BOBI user belonging to the default monitoring users group.
7. CONFIRM PASSWORD - Confirm the password by retyping it here.
8. NODE NAME – Specify the name of the BOBI node being monitored.</td></tr>
<tr><td>Outputs of the test</td><td colspan="3">One set of results for the File Repository server running in the node monitored</td></tr>
<tr><td rowspan="2">Measurements made by the test</td><td>Measurement</td><td>Measurement Unit</td><td>Interpretation</td></tr>
<tr><td>Health state:
Indicates the current health state of the File Repository server in the monitored node.</td><td></td><td>The values that this measure can report and their corresponding numeric values are discussed in the table below:
<table><tr><th>Measure Value</th><th>Numeric Value</th></tr><tr><td>Danger</td><td>0</td></tr><tr><td>Caution</td><td>1</td></tr><tr><td>Healthy</td><td>2</td></tr></table>
Note:
By default, this measure reports the Measure Values listed in the table above to indicate the health state of the server. In the graph of this measure however, the same is represented using the numeric values only.</td></tr>
</table>

<table>
<tr>
<td></td>
<td>Server running state:

Indicates the current running state of the File Repository server.</td>
<td></td>
<td>The values that this measure can report and their corresponding numeric values are discussed in the table below:

<table>
<tr><th>Measure Value</th><th>Numeric Value</th></tr>
<tr><td>STOPPED</td><td>0</td></tr>
<tr><td>STARTING</td><td>1</td></tr>
<tr><td>INITIALIZING</td><td>2</td></tr>
<tr><td>RUNNING</td><td>3</td></tr>
<tr><td>STOPPING</td><td>4</td></tr>
<tr><td>FAILED</td><td>5</td></tr>
<tr><td>RUNNING_WITH_ERRORS</td><td>6</td></tr>
<tr><td>RUNNING_WITH_WARNINGS</td><td>7</td></tr>
</table>

Note:

By default, this measure reports the Measure Values listed in the table above to indicate the running state of the server. In the graph of this measure however, the same is represented using the numeric values only.</td>
</tr>
<tr>
<td></td>
<td>Server enabled state:

Indicates whether/not the File Repository server is enabled.</td>
<td></td>
<td>The values that this measure can report and their corresponding numeric values are discussed in the table below:

<table>
<tr><th>Measure Value</th><th>Numeric Value</th></tr>
<tr><td>Disabled</td><td>0</td></tr>
<tr><td>Enabled</td><td>1</td></tr>
</table>

Note:

By default, this measure reports the Measure Values listed in the table above to indicate whether/not the server is enabled. In the graph of this measure however, the same is represented using the numeric values only.</td>
</tr>
</table>

	Busy server threads: Indicates the number of server threads that are currently servicing requests.	Number	If this measure reaches the configured maximum thread pool size for the server, new requests to the server would have to wait until a server thread becomes free. If this happens often, it may significantly slowdown request processing by the server. In such a situation, you may want to consider resizing the thread pool.

1.2.4 File Repository Server Performance Test

An Input File Repository server that is sized right and is capable of processing I/O requests quickly can alone enable users to rapidly generate new reports using the published report and program objects stored in the server. Likewise, if the Output File Repository server does not have enough space or is incapable of processing I/O requests rapidly, it cannot hold or serve completed documents for users. This is why, it would be good practice to keep an eye on the disk space usage and I/O activity on the Input and Output File Repository servers. The **File Repository Server Performance** test helps administrators do just that. This test monitors I/O requests to Input and Output File Repository servers and measures how well these servers handle these requests. In the process, the test indicates whether/not the servers are experiencing any processing bottlenecks. Additionally, the test monitors the disk space usage of the servers and proactively alerts administrators to a potential space crunch (if any) on the servers.

Purpose	Monitors I/O requests to Input and Output File Repository servers and measures how well these servers handle these requests. In the process, the test indicates whether/not the servers are experiencing any processing bottlenecks. Additionally, the test monitors the disk space usage of the servers and proactively alerts administrators to a potential space crunch (if any) on the servers.
Target of the test	A SAP BOBI node
Agent deploying the test	An internal/remote agent
Configurable parameters for the test	1. **TEST PERIOD** - How often should the test be executed 2. **HOST** - Host name of the server for which the test is to be configured 3. **PORT** - Enter the port to which the specified **HOST** listens. This should be the port at which the web application server hosting SAP BOBI listens. 4. **JMX REMOTE PORT** - Specify the RMI port number of the BOBI monitoring application. To know the RMI port number of the monitoring application, refer to Section 1.1.1 of this document. 5. **JNDI NAME** - Specify the lookup name for connecting to the JMX connector of the BOBI monitoring application. To know the JNDI name, refer to Section 1.1.1 of this document. 6. **JMX USER** and **JMX PASSWORD** – Enter the credentials of an enterprise authenticated BOBI user belonging to the default *monitoring users* group. 7. **CONFIRM PASSWORD** - Confirm the password by retyping it here. 8. **NODE NAME** – Specify the name of the BOBI node being monitored.

Outputs of the test	One set of results for the Input and Output File Repository servers in the node monitored		
Measurements made by the test	**Measurement**	**Measurement Unit**	**Interpretation**
	Active connections : Indicates the number of active connections from clients to this server.	Number	This is a good indicator of the load on the server.
	Open files: Indicates the number of files that are currently being accessed in this server.	Number	
	Data write rate: Indicates the rate at which data is written to this server.	MB/Sec	A consistent drop in the value of this measure could indicate a drop in load or a processing bottleneck.
	Data read rate: Indicates the rate at which data is read from this server.	MB/Sec	A consistent drop in the value of this measure could indicate a drop in load or a processing bottleneck.
	Free disk size: Indicates the free space on the disk containing the server's executable file.	GB	A high value is desired for this measure.
	Free disk: Indicates the percentage of free space on the disk containing the server's executable file.	Percent	A steady decrease in the value of this measure indicates that disk space on the server is getting depleted. This is a cause for concern.

1.3 The BOBI Processing Layer

Using the tests mapped to this layer, administrators can receive real-time updates on the current status of critical servers running in the monitored node, receive an overview of the load and performance of these servers, and proactively detect bottlenecks in processing in a server.

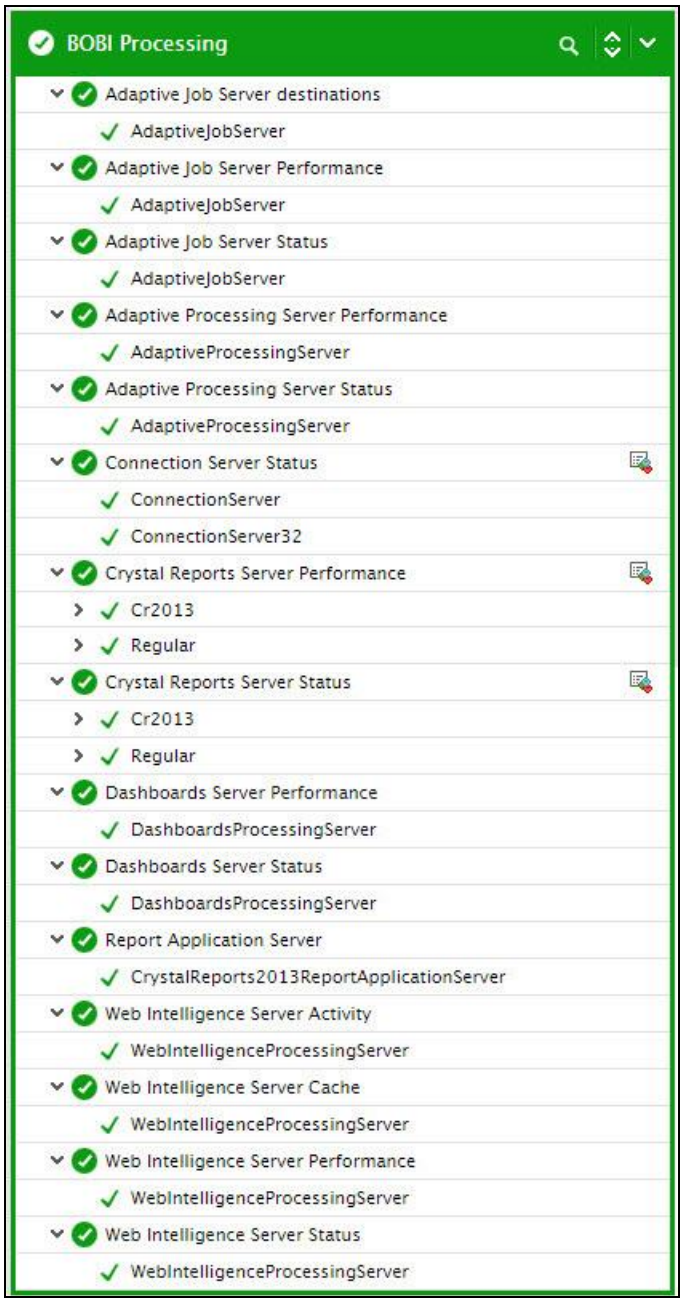

Figure 1. 9: The tests mapped to the BOBI Processing layer

1.3.1 Adaptive Job Server Status Test

The Adaptive Job Server is a "generic" server that processes Scheduled requests for different Object types. If this server stops suddenly, encounters critical errors, or slows down unexpectedly, scheduled requests will no longer be processed promptly, affecting the operations of the different object types. To avert such unpleasant eventualities, administrators must detect problems with the availability, overall health, running state, and server size of the Adaptive Job server well before end-users notice and complain. For this purpose, administrators can periodically run the **Adaptive Job Server Status** test. This test keeps an eye on the current health and status of the Adaptive Job server and alerts administrators at the first sign of an abnormality. In addition, the test also reports the count of

threads currently in use, thus providing administrators with early pointers to a potential processing bottleneck on the server.

Purpose	Keeps an eye on the current health and status of the Adaptive Job server and alerts administrators at the first sign of an abnormality. In addition, the test also reports the count of threads currently in use, thus providing administrators with early pointers to a potential processing bottleneck on the server.		
Target of the test	A SAP BOBI node		
Agent deploying the test	An internal/remote agent		
Configurable parameters for the test	1. **TEST PERIOD** - How often should the test be executed 2. **HOST** - Host name of the server for which the test is to be configured 3. **PORT** - Enter the port to which the specified **HOST** listens. This should be the port at which the web application server hosting SAP BOBI listens. 4. **JMX REMOTE PORT** - Specify the RMI port number of the BOBI monitoring application. To know the RMI port number of the monitoring application, refer to Section 1.1.1 of this document. 5. **JNDI NAME** - Specify the lookup name for connecting to the JMX connector of the BOBI monitoring application. To know the JNDI name, refer to Section 1.1.1 of this document. 6. **JMX USER** and **JMX PASSWORD** – Enter the credentials of an enterprise authenticated BOBI user belonging to the default *monitoring users* group. 7. **CONFIRM PASSWORD** - Confirm the password by retyping it here. 8. **NODE NAME** – Specify the name of the BOBI node being monitored. 9. **DETAILED DIAGNOSIS** - To make diagnosis more efficient and accurate, the eG Enterprise suite embeds an optional detailed diagnostic capability. With this capability, the eG agents can be configured to run detailed, more elaborate tests as and when specific problems are detected. To enable the detailed diagnosis capability of this test for a particular server, choose the On option. To disable the capability, click on the Off option. The option to selectively enable/disable the detailed diagnosis capability will be available only if the following conditions are fulfilled: • The eG manager license should allow the detailed diagnosis capability • Both the normal and abnormal frequencies configured for the detailed diagnosis measures should not be 0.		
Outputs of the test	One set of results for Adaptive Job server running in the node monitored		
Measurements made by the	**Measurement**	**Measurement Unit**	**Interpretation**

<table>
<tr>
<td>test</td>
<td>Health state:

Indicates the current health state of the Adaptive Job server in the monitored node.</td>
<td></td>
<td>The values that this measure can report and their corresponding numeric values are discussed in the table below:

<table>
<tr><th>Measure Value</th><th>Numeric Value</th></tr>
<tr><td>Danger</td><td>0</td></tr>
<tr><td>Caution</td><td>1</td></tr>
<tr><td>Healthy</td><td>2</td></tr>
</table>

Note:

By default, this measure reports the Measure Values listed in the table above to indicate the health state of the server. In the graph of this measure however, the same is represented using the numeric values only.</td>
</tr>
<tr>
<td></td>
<td>Server running state:

Indicates the current running state of the Adaptive Job server.</td>
<td></td>
<td>The values that this measure can report and their corresponding numeric values are discussed in the table below:

<table>
<tr><th>Measure Value</th><th>Numeric Value</th></tr>
<tr><td>STOPPED</td><td>0</td></tr>
<tr><td>STARTING</td><td>1</td></tr>
<tr><td>INITIALIZING</td><td>2</td></tr>
<tr><td>RUNNING</td><td>3</td></tr>
<tr><td>STOPPING</td><td>4</td></tr>
<tr><td>FAILED</td><td>5</td></tr>
<tr><td>RUNNING_WITH_ERRORS</td><td>6</td></tr>
<tr><td>RUNNING_WITH_WARNINGS</td><td>7</td></tr>
</table>

If the server is not healthy, the detailed diagnosis of this measure, if enabled, provides the process ID of the server.

Note:

By default, this measure reports the Measure Values listed in the table above to indicate the running state of the server. In the graph of this measure however, the same is represented using the numeric values only.</td>
</tr>
</table>

<table>
<tr>
<td></td>
<td>Server enabled state:
Indicates whether/not the Adaptive Job server is enabled.</td>
<td></td>
<td>The values that this measure can report and their corresponding numeric values are discussed in the table below:
<table>
<tr><th>Measure Value</th><th>Numeric Value</th></tr>
<tr><td>Disabled</td><td>0</td></tr>
<tr><td>Enabled</td><td>1</td></tr>
</table>
Note:
By default, this measure reports the Measure Values listed in the table above to indicate whether/not the server is enabled. In the graph of this measure however, the same is represented using the numeric values only.</td>
</tr>
<tr>
<td></td>
<td>Busy server threads:
Indicates the number of server threads that are currently servicing requests.</td>
<td>Number</td>
<td>If this measure reaches the configured maximum thread pool size for the server, new requests to the server would have to wait until a server thread becomes free. If this happens often, it may significantly slowdown request processing by the server. In such a situation, you may want to consider resizing the thread pool.</td>
</tr>
</table>

1.3.2 Adaptive Job Server Destinations Test

When you add a Job server to the SAP BusinessObjects Business Intelligence platform system, you can configure the Job server to process reports, documents, programs, or publications and send the results to different destinations, including file systems, and email, or accessed through web sites or portals. If the Job Server is not able to send reports/documents to a particular destination, it could be because the destination specification is invalid/unreacheable. To isolate such invalid/inaccessible destinations, use the **Adaptive Job Server Destinations** test. This test checks whether/not the Job server is able to send documents to a destination, and if not, reports that destination as invalid or unreachable.

Purpose	Checks whether/not the Job server is able to send documents to a destination, and if not, reports that destination as invalid or unreachable
Target of the test	A SAP BOBI node
Agent deploying the test	An internal/remote agent

<table>
<tr><td>Configurable parameters for the test</td><td colspan="3">1. TEST PERIOD - How often should the test be executed
2. HOST - Host name of the server for which the test is to be configured
3. PORT - Enter the port to which the specified HOST listens. This should be the port at which the web application server hosting SAP BOBI listens.
4. JMX REMOTE PORT - Specify the RMI port number of the BOBI monitoring application. To know the RMI port number of the monitoring application, refer to Section 1.1.1 of this document.
5. JNDI NAME - Specify the lookup name for connecting to the JMX connector of the BOBI monitoring application. To know the JNDI name, refer to Section 1.1.1 of this document.
6. JMX USER and JMX PASSWORD – Enter the credentials of an enterprise authenticated BOBI user belonging to the default monitoring users group.
7. CONFIRM PASSWORD - Confirm the password by retyping it here.
8. NODE NAME – Specify the name of the BOBI node being monitored.</td></tr>
<tr><td>Outputs of the test</td><td colspan="3">One set of results for the Adaptive Job server running in the node monitored</td></tr>
<tr><td>Measurements made by the test</td><td>Measurement</td><td>Measurement Unit</td><td>Interpretation</td></tr>
<tr><td></td><td>Is file system destination valid ?:
Indicates whether/not the server is able to send documents to the file system destination configured for it.</td><td></td><td>The values that this measure can report and their corresponding numeric values are discussed in the table below:
<table><tr><th>Measure Value</th><th>Numeric Value</th></tr><tr><td>True</td><td>100</td></tr><tr><td>False</td><td>0</td></tr></table>
Note:
By default, this measure reports the Measure Values listed in the table above to indicate whether/not the Job server is able to send documents to a configured destination. In the graph of this measure however, the same is represented using the numeric values only.</td></tr>
</table>

<table>
<tr>
<td></td>
<td>Is FTP destination valid?:

Indicates whether/not the server is able to send documents to the FTP destination configured for it.</td>
<td></td>
<td>The values that this measure can report and their corresponding numeric values are discussed in the table below:

<table><tr><th>Measure Value</th><th>Numeric Value</th></tr><tr><td>True</td><td>100</td></tr><tr><td>False</td><td>0</td></tr></table>
Note:

By default, this measure reports the Measure Values listed in the table above to indicate whether/not the Job server is able to send documents to a configured destination. In the graph of this measure however, the same is represented using the numeric values only.</td>
</tr>
<tr>
<td></td>
<td>Is Inbox destination valid?:

Indicates whether/not the server is able to send documents to the Inbox destination configured for it.</td>
<td></td>
<td>The values that this measure can report and their corresponding numeric values are discussed in the table below:

<table><tr><th>Measure Value</th><th>Numeric Value</th></tr><tr><td>True</td><td>100</td></tr><tr><td>False</td><td>0</td></tr></table>
Note:

By default, this measure reports the Measure Values listed in the table above to indicate whether/not the Job server is able to send documents to a configured destination. In the graph of this measure however, the same is represented using the numeric values only.</td>
</tr>
</table>

<table>
<tr>
<td></td>
<td>Is Email destination valid?:

Indicates whether/not the server is able to send documents to the email destination configured for it.</td>
<td></td>
<td>The values that this measure can report and their corresponding numeric values are discussed in the table below:

<table>
<tr><th>Measure Value</th><th>Numeric Value</th></tr>
<tr><td>True</td><td>100</td></tr>
<tr><td>False</td><td>0</td></tr>
</table>

Note:

By default, this measure reports the Measure Values listed in the table above to indicate whether/not the Job server is able to send documents to a configured destination. In the graph of this measure however, the same is represented using the numeric values only.</td>
</tr>
</table>

	Is SAP Streamwork destination valid?: Indicates whether/not the server is able to send documents to the SAP StreamWork destination configured for it.		SAP StreamWork is an enterprise collaboration tool, which allows real-time collaboration of business activities such as analyzing data, planning meetings, and making decisions. It incorporates technology from Box.net and Evernote to allow users to connect to online files and documents, and document-reader technology from Scribd to allow users to view documents directly within its environment. One of the key advantages of integrating SAP BOBI with SAP StreamWork is that Crystal reports and Web Intelligence documents, which are a product of the SAP BOBI system, can be sent to SAP StreamWork for viewing and analysis. If, post this integration, the Job server is not able to send documents to SAP StreamWork, then the value of this measure will be *False*. If documents are successfully sent to SAP StreamWork, the value of this measure will be *True*. The numeric values that correspond to these measure values are discussed in the table below: **Measure Value** / **Numeric Value** True / 100 False / 0 **Note:** By default, this measure reports the **Measure Value**s listed in the table above to indicate whether/not the Job server is able to send documents to a configured destination. In the graph of this measure however, the same is represented using the numeric values only.

1.3.3 Adaptive Job Server Performance Test

The Adaptive Job server is a general server that processes scheduled jobs. A processing bottleneck in this server can therefore significantly slowdown the rate at which the server processes programs, documents, reports, and publications. Also, errors affecting server operations can also cause jobs to fail frequently. To ensure uninterrupted job processing and a high success rate for jobs, administrators should continuously track the number and status of jobs processed by the Job server, proactively detect potential processing bottlenecks, and rapidly initiate measures to eliminate these bottlenecks. This is where the **Adaptive Job Server Performance** test is most useful! This test closely tracks the job requests received by the server and reports the count of requested jobs that are being processed

currently and those that failed. This way, the test signals a probable processing delay or processing error, and thus enables administrators to initiate pre-emptive measures soon.

Purpose	Closely tracks the job requests received by the server and reports the count of requested jobs that are being processed currently and those that failed. This way, the test signals a probable processing delay or processing error, and thus enables administrators to initiate pre-emptive measures soon.		
Target of the test	A SAP BOBI node		
Agent deploying the test	An internal/remote agent		
Configurable parameters for the test	1. **TEST PERIOD** - How often should the test be executed 2. **HOST** - Host name of the server for which the test is to be configured 3. **PORT** - Enter the port to which the specified **HOST** listens. This should be the port at which the web application server hosting SAP BOBI listens. 4. **JMX REMOTE PORT** - Specify the RMI port number of the BOBI monitoring application. To know the RMI port number of the monitoring application, refer to Section 1.1.1 of this document. 5. **JNDI NAME** - Specify the lookup name for connecting to the JMX connector of the BOBI monitoring application. To know the JNDI name, refer to Section 1.1.1 of this document. 6. **JMX USER** and **JMX PASSWORD** – Enter the credentials of an enterprise authenticated BOBI user belonging to the default *monitoring users* group. 7. **CONFIRM PASSWORD** - Confirm the password by retyping it here. 8. **NODE NAME** – Specify the name of the BOBI node being monitored.		
Outputs of the test	One set of results for the Adaptive Job server running in the node monitored		
Measurements made by the test	**Measurement**	**Measurement Unit**	**Interpretation**
	Received job requests: Indicates the nusmber of job requests received for processing in the server.	Number	This is a good indicator of the current request load on the server.

	Concurrent jobs: Indicates.the number of concurrent jobs currently running in the server.	Number	The Job server is configured to process a maximum of 5 concurrent jobs per service by default. Any additional jobs will remain pending, until the completion of existing jobs. If the value of this measure is very low, but the number of job requests to the server is very high, you many want to increase the concurrent job configuration, so that the server is able to process more jobs concurrently. If the server still chokes, check the *Busy server threads* measure reported by the **Adaptive Job Server Status** test to see how the thread pool is utilized currently. If it is being used up to capacity, consider increasing the thread pool size to meet with the demand.
	Failed job creations: Indicates the number of jobs that have failed on the server.	Number	Ideally, the value of this measure should be 0. A high value is a cause for concern, as it indicates that too many jobs are failing.

1.3.4 Adaptive Processing Server Status Test

The Adaptive Processing Server (APS) is a "generic" server that processes non-Object / post-processing requests. It hosts a lot of the BI services. Multiple APSes may be defined on multiple nodes within a deployment. In almost all cases more than one APS will be needed in the system, both for management and maintenance of the running services.

If the APS fails, experiences errors, or is unable to process requests as quickly as they are received, the services it hosts will not be able to function properly, thus impacting user productivity. To prevent this outcome, administrators must rapidly detect issues affecting the health, status, and processing ability of the APS, and swiftly initiate measures to resolve them. This can be achieved with the help of the **Adaptive Processing Server Status** test. This test keeps tabs on the health, status, and processing ability of the APS, and warns administrators of impending dangers to the availability and overall performance of the server.

Purpose	Keeps tabs on the health, status, and processing ability of the APS, and warns administrators of impending dangers to the availability and overall performance of the server.
Target of the test	A SAP BOBI node
Agent deploying the test	An internal/remote agent

<table>
<tr><td>Configurable parameters for the test</td><td colspan="3">1. TEST PERIOD - How often should the test be executed
2. HOST - Host name of the server for which the test is to be configured
3. PORT - Enter the port to which the specified HOST listens. This should be the port at which the web application server hosting SAP BOBI listens.
4. JMX REMOTE PORT - Specify the RMI port number of the BOBI monitoring application. To know the RMI port number of the monitoring application, refer to Section 1.1.1 of this document.
5. JNDI NAME - Specify the lookup name for connecting to the JMX connector of the BOBI monitoring application. To know the JNDI name, refer to Section 1.1.1 of this document.
6. JMX USER and JMX PASSWORD – Enter the credentials of an enterprise authenticated BOBI user belonging to the default monitoring users group.
7. CONFIRM PASSWORD - Confirm the password by retyping it here.
8. NODE NAME – Specify the name of the BOBI node being monitored.
9. DETAILED DIAGNOSIS - To make diagnosis more efficient and accurate, the eG Enterprise suite embeds an optional detailed diagnostic capability. With this capability, the eG agents can be configured to run detailed, more elaborate tests as and when specific problems are detected. To enable the detailed diagnosis capability of this test for a particular server, choose the On option. To disable the capability, click on the Off option.
The option to selectively enable/disable the detailed diagnosis capability will be available only if the following conditions are fulfilled:
• The eG manager license should allow the detailed diagnosis capability
• Both the normal and abnormal frequencies configured for the detailed diagnosis measures should not be 0.</td></tr>
<tr><td>Outputs of the test</td><td colspan="3">One set of results for Adaptive Processing server running in the node monitored</td></tr>
<tr><td>Measurements made by the</td><td>Measurement</td><td>Measurement Unit</td><td>Interpretation</td></tr>
</table>

<table>
<tr>
<td>test</td>
<td>Health state:

Indicates the current health state of the Adaptive Processing server in the monitored node.</td>
<td></td>
<td>The values that this measure can report and their corresponding numeric values are discussed in the table below:
<table><tr><th>Measure Value</th><th>Numeric Value</th></tr><tr><td>Danger</td><td>0</td></tr><tr><td>Caution</td><td>1</td></tr><tr><td>Healthy</td><td>2</td></tr></table>
If the server is not healthy, then the detailed diagnosis of this measure, if enabled, provides the process ID, JVM debug information, trace flags, JVM version and service hosted by the server.

Note:

By default, this measure reports the Measure Values listed in the table above to indicate the health state of the server. In the graph of this measure however, the same is represented using the numeric values only.</td>
</tr>
<tr>
<td></td>
<td>Server running state:

Indicates the current running state of the Adaptive Processing server.</td>
<td></td>
<td>The values that this measure can report and their corresponding numeric values are discussed in the table below:
<table><tr><th>Measure Value</th><th>Numeric Value</th></tr><tr><td>STOPPED</td><td>0</td></tr><tr><td>STARTING</td><td>1</td></tr><tr><td>INITIALIZING</td><td>2</td></tr><tr><td>RUNNING</td><td>3</td></tr><tr><td>STOPPING</td><td>4</td></tr><tr><td>FAILED</td><td>5</td></tr><tr><td>RUNNING_WITH_ERRORS</td><td>6</td></tr><tr><td>RUNNING_WITH_WARNINGS</td><td>7</td></tr></table>
Note:

By default, this measure reports the Measure Values listed in the table above to indicate the running state of the server. In the graph of this measure however, the same is represented using the numeric values only.</td>
</tr>
</table>

<table>
<tr>
<td></td>
<td>Server enabled state:
Indicates whether/not the Adaptive Processing server is enabled.</td>
<td></td>
<td>The values that this measure can report and their corresponding numeric values are discussed in the table below:
<table>
<tr><th>Measure Value</th><th>Numeric Value</th></tr>
<tr><td>Disabled</td><td>0</td></tr>
<tr><td>Enabled</td><td>1</td></tr>
</table>
Note:
By default, this measure reports the Measure Values listed in the table above to indicate whether/not the server is enabled. In the graph of this measure however, the same is represented using the numeric values only.</td>
</tr>
<tr>
<td></td>
<td>Busy server threads:
Indicates the number of server threads that are currently servicing requests.</td>
<td>Number</td>
<td>If this measure reaches the configured maximum thread pool size for the server, new requests to the server would have to wait until a server thread becomes free. If this happens often, it may significantly slowdown request processing by the server. In such a situation, you may want to consider resizing the thread pool.</td>
</tr>
</table>

1.3.5 Adaptive Processing Server Performance Test

One of the key factors influencing the performance of the Adaptive Processing Server (APS) is the usage of its JVM memory heap. This is because, APS is a pure Java based process, initialized with 1 GB of Java Heap. Naturally therefore, the lack of adequate free memory to the JVM, faulty and frequent garbage collections, and JVM deadlocks can all have an adverse impact on the health of the APS. Likewise, if critical services hosted on the APS – such as the BEx web application server and the Client auditing proxy server – are not correctly configured to handle the requests they receive, then again APS performance will degrade. This is why, the eG agent periodically runs the **Adaptive Processing Server Performance** test. This test enables administrators measure JVM health and the correctness of the configuration of the critical APS services, thus helping them rapidly detect dips in APS performance and the possible reasons for it.

Purpose	Enables administrators measure JVM health and the correctness of the configuration of the critical APS services, thus helping them rapidly detect dips in APS performance and the possible reasons for it
Target of the test	A SAP BOBI node
Agent deploying the test	An internal/remote agent

Configurable parameters for the test	1. **TEST PERIOD** - How often should the test be executed 2. **HOST** - Host name of the server for which the test is to be configured 3. **PORT** - Enter the port to which the specified **HOST** listens. This should be the port at which the web application server hosting SAP BOBI listens. 4. **JMX REMOTE PORT** - Specify the RMI port number of the BOBI monitoring application. To know the RMI port number of the monitoring application, refer to Section 1.1.1 of this document. 5. **JNDI NAME** - Specify the lookup name for connecting to the JMX connector of the BOBI monitoring application. To know the JNDI name, refer to Section 1.1.1 of this document. 6. **JMX USER** and **JMX PASSWORD** – Enter the credentials of an enterprise authenticated BOBI user belonging to the default *monitoring users* group. 7. **CONFIRM PASSWORD** - Confirm the password by retyping it here. 8. **NODE NAME** – Specify the name of the BOBI node being monitored.		
Outputs of the test	One set of results for the APS running in the node monitored		
Measurements made by the test	**Measurement**	**Measurement Unit**	**Interpretation**
	Free JVM memory: Indicates the amount of memory available to the JVM for allocating new objects.	GB	Ideally, the value of this measure should be high.
	Free JVM memory percentage: Indicates the percentage of memory available to the JVM for allocating new objects.	Percent	A value close to 100% is a cause for concern, as it indicates rapid erosion of the JVM memory heap. Without sufficient memory, the APS and its services will not be able to operate optimally.
	CPU usage in last 5 mins: Indicates the percentage of time the CPU was used by the APS during the last 5 mins.	Percent	This measure considers all processors allocated to the JVM. A value close to 100% indicates excessive CPU usage, probably owing to CPU-intensive operations performed on the JVM. If more processing power is not allocated to the JVM, the APS may hang.
	Stopped system time during GC in last 5 mins: Indicates the percentage of time that APS services were stopped for Garbage Collection in the last 5 minutes.	Percent	A critical stage of garbage collection requires exclusive access and all APS services are halted at this time. This value should always be less than 10. 10 and above indicates a low throughput issue and requires further investigation

	Number of page faults during GC in last 5 mins: Indicates the number of page faults that occurred while garbage collection was running during the last five minutes.	Number	Any value greater than 0 indicates a system under heavy load and low memory conditions.
	JVM lock contention lock: Indicates the current number of JVM lock contentions.	Number	This represents the number of synchronized objects that have threads that are waiting for access. The average value of this measure should be 0. Consistently higher values indicates threads that will not run again. You may want to take a thread dump to investigate such issues.
	JVM deadlocked threads counter: Indicates the number of active sessions within a Business Explorer Web Applications Service.	Number	These threads are indefinitely waiting on each other for a common set of resources. Average value should be 0. Consistently higher values warranties further investigation using thread dumps
	BEx web application service sessions: Indicates the number of active sessions within a Business Explorer Web Applications Service.	Number	BEx Web applications are Web-based applications from the Business Explorer (BEx) of SAP NetWeaver Business Warehouse (BW) for data analysis, reporting, and analytical applications on the Web. This measure is a good indicator of the load generated by this service on the APS.
	Client audit event rate: Indicates the rate of client auditing events received within the last measure period.	Events/Sec	If the event is configured to be audited, the client sends the event information to the web application server, which passes it to the Client Auditing Proxy Service (CAPS) hosted in an Adaptive Processing Server (APS). This measure is used to monitor the configuration and load on the Client auditing proxy service. Values greater than 0 indicate that the service has been configured correctly.

	Current number of auditing events queued: Indicates the number of auditing events that the APS has recorded, but which have not yet been retrieved by the CMS Auditor.	Number	If this number increases without bound, it could mean indicate that auditing has not been configured properly or that the system is heavily loaded and generating auditing events faster than the auditor can retrieve them. When stopping servers, It is advisable to disable them first and wait for auditing events to be fully retrieved and this queue becomes empty. Otherwise, they may be retrieved only when this server has been restarted and the CMS polls for them.

1.3.6 Connection Server Status Test

The Connection Server provides database access to source data. It supports relational databases, as well as OLAP and other formats. The Connection Server is responsible for handling connection and interaction with the various data sources and providing a common feature set to clients. Without the connection server therefore, source data cannot be accessed, thereby causing critical services to fail. Moreover, if the connection server slows down, so will database accesses. To avoid such failures/slowdowns, administrators should run the **Connection Server Status** test at regular intervals, check the status of the connection server, determine whether/not the server is sized right to process its current and future request load, and in the process, promptly detect a server failure or slowness. Quick and accurate problem identification will enable administrators to swiftly initiate corrective action, so that the Connection server resumes normal operations in no time.

Purpose	Checks the status of the connection server, reveals whether/not the server is sized right to process its current and future request load, and in the process, helps administrators promptly detect a server failure or slowness
Target of the test	A SAP BOBI node
Agent deploying the test	An internal/remote agent

Configurable parameters for the test	1. **TEST PERIOD** - How often should the test be executed 2. **HOST** - Host name of the server for which the test is to be configured 3. **PORT** - Enter the port to which the specified **HOST** listens. This should be the port at which the web application server hosting SAP BOBI listens. 4. **JMX REMOTE PORT** - Specify the RMI port number of the BOBI monitoring application. To know the RMI port number of the monitoring application, refer to Section 1.1.1 of this document. 5. **JNDI NAME** - Specify the lookup name for connecting to the JMX connector of the BOBI monitoring application. To know the JNDI name, refer to Section 1.1.1 of this document. 6. **JMX USER** and **JMX PASSWORD** – Enter the credentials of an enterprise authenticated BOBI user belonging to the default *monitoring users* group. 7. **CONFIRM PASSWORD** - Confirm the password by retyping it here. 8. **NODE NAME** – Specify the name of the BOBI node being monitored. 9. **DETAILED DIAGNOSIS** - To make diagnosis more efficient and accurate, the eG Enterprise suite embeds an optional detailed diagnostic capability. With this capability, the eG agents can be configured to run detailed, more elaborate tests as and when specific problems are detected. To enable the detailed diagnosis capability of this test for a particular server, choose the On option. To disable the capability, click on the Off option. The option to selectively enable/disable the detailed diagnosis capability will be available only if the following conditions are fulfilled: • The eG manager license should allow the detailed diagnosis capability • Both the normal and abnormal frequencies configured for the detailed diagnosis measures should not be 0.		
Outputs of the test	One set of results for the Connection server running in the node monitored		
Measurements made by the	**Measurement**	**Measurement Unit**	**Interpretation**

<table>
<tr>
<td>test</td>
<td>Health state:

Indicates the current health state of the Connection server in the monitored node.</td>
<td></td>
<td>The values that this measure can report and their corresponding numeric values are discussed in the table below:

<table><tr><th>Measure Value</th><th>Numeric Value</th></tr><tr><td>Danger</td><td>0</td></tr><tr><td>Caution</td><td>1</td></tr><tr><td>Healthy</td><td>2</td></tr></table>
If the server is not healthy, then the detailed diagnosis of this measure, if enabled, will provide the process ID of the server.

Note:

By default, this measure reports the Measure Values listed in the table above to indicate the health state of the server. In the graph of this measure however, the same is represented using the numeric values only.</td>
</tr>
<tr>
<td></td>
<td>Server running state:

Indicates the current running state of the Connection server.</td>
<td></td>
<td>The values that this measure can report and their corresponding numeric values are discussed in the table below:

<table><tr><th>Measure Value</th><th>Numeric Value</th></tr><tr><td>STOPPED</td><td>0</td></tr><tr><td>STARTING</td><td>1</td></tr><tr><td>INITIALIZING</td><td>2</td></tr><tr><td>RUNNING</td><td>3</td></tr><tr><td>STOPPING</td><td>4</td></tr><tr><td>FAILED</td><td>5</td></tr><tr><td>RUNNING_WITH_ERRORS</td><td>6</td></tr><tr><td>RUNNING_WITH_WARNINGS</td><td>7</td></tr></table>
Note:

By default, this measure reports the Measure Values listed in the table above to indicate the running state of the server. In the graph of this measure however, the same is represented using the numeric values only.</td>
</tr>
</table>

	Server enabled state: Indicates whether/not the Connection server is enabled.		The values that this measure can report and their corresponding numeric values are discussed in the table below: **Measure Value** / **Numeric Value** Disabled / 0 Enabled / 1 **Note:** By default, this measure reports the **Measure Value**s listed in the table above to indicate whether/not the server is enabled. In the graph of this measure however, the same is represented using the numeric values only.
	Busy server threads: Indicates the number of server threads that are currently servicing requests.	Number	If this measure reaches the configured maximum thread pool size for the server, new requests to the server would have to wait until a server thread becomes free. If this happens often, it may significantly slowdown request processing by the server. In such a situation, you may want to consider resizing the thread pool.

1.3.7 Crystal Reports Server Status Test

The Crystal Reports Processing Server responds 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 only the requested page is returned, not the entire report. This improves system performance and reduces unnecessary network traffic for large reports. If this server is not running or is slow in processing page requests, EPF pages will either not be generated at all or may take hours to be generated. As a result, report generation will take longer than normal, thereby adversely impacting user experience with the reporting tool. This is why, it is imperative that administrators are notified of even the slightest deviation in the availability and performance of the Crystal Reports Processing server. This is exactly what the **Crystal Reports Server Status** test does. This test sends out email/SMS alerts to users when the Crystal Reports Processing server switches to an abnormal state suddenly or when it runs out of free threads for processing subsequent page requests. In the process, the test makes sure that the administrator promptly addresses these issues and quickly restores normalcy.

Purpose	Sends out email/SMS alerts to users when the Crystal Reports Processing server switches to an abnormal state suddenly or when it runs out of free threads for processing subsequent page requests. In the process, the test makes sure that the administrator promptly address these issues and quickly restore normalcy.
Target of the test	A SAP BOBI node
Agent deploying the	An internal/remote agent

<table>
<tr><td>test</td><td colspan="3"></td></tr>
<tr><td>Configurable parameters for the test</td><td colspan="3">1. TEST PERIOD - How often should the test be executed
2. HOST - Host name of the server for which the test is to be configured
3. PORT - Enter the port to which the specified HOST listens. This should be the port at which the web application server hosting SAP BOBI listens.
4. JMX REMOTE PORT - Specify the RMI port number of the BOBI monitoring application. To know the RMI port number of the monitoring application, refer to Section 1.1.1 of this document.
5. JNDI NAME - Specify the lookup name for connecting to the JMX connector of the BOBI monitoring application. To know the JNDI name, refer to Section 1.1.1 of this document.
6. JMX USER and JMX PASSWORD – Enter the credentials of an enterprise authenticated BOBI user belonging to the default monitoring users group.
7. CONFIRM PASSWORD - Confirm the password by retyping it here.
8. NODE NAME – Specify the name of the BOBI node being monitored.
9. DETAILED DIAGNOSIS - To make diagnosis more efficient and accurate, the eG Enterprise suite embeds an optional detailed diagnostic capability. With this capability, the eG agents can be configured to run detailed, more elaborate tests as and when specific problems are detected. To enable the detailed diagnosis capability of this test for a particular server, choose the On option. To disable the capability, click on the Off option.
The option to selectively enable/disable the detailed diagnosis capability will be available only if the following conditions are fulfilled:
• The eG manager license should allow the detailed diagnosis capability
• Both the normal and abnormal frequencies configured for the detailed diagnosis measures should not be 0.</td></tr>
<tr><td>Outputs of the test</td><td colspan="3">One set of results for the Crystal Reports Processing server running in the node monitored</td></tr>
<tr><td>Measurements made by the</td><td>Measurement</td><td>Measurement Unit</td><td>Interpretation</td></tr>
</table>

<table>
<tr>
<td>test</td>
<td>Health state:

Indicates the current health state of the Crystal Reports Processing server in the monitored node.</td>
<td></td>
<td>The values that this measure can report and their corresponding numeric values are discussed in the table below:

<table>
<tr><th>Measure Value</th><th>Numeric Value</th></tr>
<tr><td>Danger</td><td>0</td></tr>
<tr><td>Caution</td><td>1</td></tr>
<tr><td>Healthy</td><td>2</td></tr>
</table>

If the server is not healthy, then the detailed diagnosis of this measure, if enabled, provides the process ID and processing plugin name of the server.

Note:

By default, this measure reports the Measure Values listed in the table above to indicate the health state of the server. In the graph of this measure however, the same is represented using the numeric values only.</td>
</tr>
<tr>
<td></td>
<td>Server running state:

Indicates the current running state of the Crystal Reports Processing server.</td>
<td></td>
<td>The values that this measure can report and their corresponding numeric values are discussed in the table below:

<table>
<tr><th>Measure Value</th><th>Numeric Value</th></tr>
<tr><td>STOPPED</td><td>0</td></tr>
<tr><td>STARTING</td><td>1</td></tr>
<tr><td>INITIALIZING</td><td>2</td></tr>
<tr><td>RUNNING</td><td>3</td></tr>
<tr><td>STOPPING</td><td>4</td></tr>
<tr><td>FAILED</td><td>5</td></tr>
<tr><td>RUNNING_WITH_ERRORS</td><td>6</td></tr>
<tr><td>RUNNING_WITH_WARNINGS</td><td>7</td></tr>
</table>

Note:

By default, this measure reports the Measure Values listed in the table above to indicate the running state of the server. In the graph of this measure however, the same is represented using the numeric values only.</td>
</tr>
</table>

	Server enabled state: Indicates whether/not the Crystal Reports Processing server is enabled.		The values that this measure can report and their corresponding numeric values are discussed in the table below: **Measure Value** / **Numeric Value** Disabled / 0 Enabled / 1 **Note:** By default, this measure reports the **Measure Value**s listed in the table above to indicate whether/not the server is enabled. In the graph of this measure however, the same is represented using the numeric values only.
	Busy server threads: Indicates the number of server threads that are currently servicing requests.	Number	If this measure reaches the configured maximum thread pool size for the server, new requests to the server would have to wait until a server thread becomes free. If this happens often, it may significantly slowdown request processing by the server. In such a situation, you may want to consider resizing the thread pool.

1.3.8 Crystal Reports Server Performance Test

A healthy Crystal Reports Processing Server is one that processes all the page requests it receives without fail and in minimum time, so that EPF pages are quickly generated and returned to clients. If requests to a Crystal Reports, Processing Server fail often or take too long to be processed, EPF page generation will be considerably delayed. As a result, the server's promise of instant, page-on-demand access to users will stand broken, killing user confidence in its abilities. Moreover, any slowdown in request processing will also result in long pending request queues, which if allowed to grow may choke the server. In short, processing bottlenecks with the Crystal Reports Processing Server can prove to be detrimental to the health of the server and will also damage user experience with the server. This is why, it is imperative that administrators identify a probable slowdown or outage of the server at its nascent stages itself, so that they can avert such fatalities well before they occur. The **Crystal Reports Server Performance** test helps administrators with this.

This test closely tracks page requests to the server, continuously observes the rate at which the server processes page requests, rapidly isolates probable failures/slowdowns, and promptly alerts administrators to them. This way, the test provides administrators with a heads-up on potentially serious processing bottlenecks, so that administrators can work towards preventing them.

Purpose	Cllosely tracks page requests to the server, continuously observes the rate at which the server processes page requests, rapidly isolates failures/slowdowns, and promptly alerts administrators to them. This way, the test provides administrators with a heads-up on potentially serious processing bottlenecks, so that administrators can work towards preventing them

Target of the test	A SAP BOBI node		
Agent deploying the test	An internal/remote agent		
Configurable parameters for the test	1. **TEST PERIOD** - How often should the test be executed 2. **HOST** - Host name of the server for which the test is to be configured 3. **PORT** - Enter the port to which the specified **HOST** listens. This should be the port at which the web application server hosting SAP BOBI listens. 4. **JMX REMOTE PORT** - Specify the RMI port number of the BOBI monitoring application. To know the RMI port number of the monitoring application, refer to Section 1.1.1 of this document. 5. **JNDI NAME** - Specify the lookup name for connecting to the JMX connector of the BOBI monitoring application. To know the JNDI name, refer to Section 1.1.1 of this document. 6. **JMX USER** and **JMX PASSWORD** – Enter the credentials of an enterprise authenticated BOBI user belonging to the default *monitoring users* group. 7. **CONFIRM PASSWORD** - Confirm the password by retyping it here. 8. **NODE NAME** – Specify the name of the BOBI node being monitored.		
Outputs of the test	One set of results for the Crystal Reports Processing server running in the node monitored		
Measurements made by the test	**Measurement**	**Measurement Unit**	**Interpretation**
	Processing rate: Indicates the rate at which the server processes the page requests it receives.	Requests/Sec	A consistent drop in the value of this measure could indicate a probable processing slowdown.
	Number of open jobs: Indicates the number of jobs that the server and its child processes are currently processing.	Number	This is a good indicator of the current workload of the server.
	Average processing time: Indicates the average time taken by the server to process a request (taken over the last 500 requests).	Millisecs	Ideally, the value of this measure should be low. A high value or a consistent rise in this value is a cause for concern, as it indicates a slowdown in request processing.

	Number of queued requests: Indicates the number of requests waiting to be processed or are being processed.	Number	A steady rise in the value of this measure is a sign of a request queue that is growing. This means that the server is unable to process page requests as quickly as it receives them. You may want to check the value of the *Busy server threads* measure of the **Crystal Reports Processing Server Status** test to know if the lack of free threads in the thread pool is the reason for the slowness.
	Number of open connections: Indicates the number of connections open between the server and its clients.	Number	
	Percentage requests failed: Indicates what percentage of the last 500 requests have failed.	Percent	Ideally, the value of this measure should be 0. A non-zero value is a sign of trouble. A value close to 100 could hint at a critical problem condition that warrants an immediate and thorough investigation.
	Data transfer rate: Indicates the rate at which data is transferred from the server to its clients in the last measure period.	KB/Sec	

1.3.9 Dashboards Server Status Test

The Dashboards Processing Server responds to Dashboard 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 only the requested page is returned, not the entire report. This improves system performance and reduces unnecessary network traffic for large reports.

If this server is not running or is slow in processing dashboard requests, EPF pages will either not be generated at all or may take hours to be generated. As a result, report generation will take longer than normal, thereby adversely impacting user experience with dashboards. This is why, it is imperative that administrators are notified of even the slightest deviation in the availability and performance of the Dashboards Processing server. This is exactly what the **Dashboards Server Status** test does. This test sends out email/SMS alerts to users when the Dashboards Processing server switches to an abnormal state suddenly or when it runs out of free threads for processing subsequent dashboard requests. In the process, the test makes sure that the administrator promptly addresses these issues and quickly restores normalcy.

Purpose	Sends out email/SMS alerts to users when the Dashboards Processing server switches to an abnormal state suddenly or when it runs out of free threads for processing subsequent dashboard requests. In the process, the test makes sure that the administrator promptly addresses these issues and quickly restores normalcy

Target of the test	A SAP BOBI node		
Agent deploying the test	An internal/remote agent		
Configurable parameters for the test	1. **TEST PERIOD** - How often should the test be executed 2. **HOST** - Host name of the server for which the test is to be configured 3. **PORT** - Enter the port to which the specified **HOST** listens. This should be the port at which the web application server hosting SAP BOBI listens. 4. **JMX REMOTE PORT** - Specify the RMI port number of the BOBI monitoring application. To know the RMI port number of the monitoring application, refer to Section 1.1.1 of this document. 5. **JNDI NAME** - Specify the lookup name for connecting to the JMX connector of the BOBI monitoring application. To know the JNDI name, refer to Section 1.1.1 of this document. 6. **JMX USER** and **JMX PASSWORD** – Enter the credentials of an enterprise authenticated BOBI user belonging to the default *monitoring users* group. 7. **CONFIRM PASSWORD** - Confirm the password by retyping it here. 8. **NODE NAME** – Specify the name of the BOBI node being monitored. 9. **DETAILED DIAGNOSIS** - To make diagnosis more efficient and accurate, the eG Enterprise suite embeds an optional detailed diagnostic capability. With this capability, the eG agents can be configured to run detailed, more elaborate tests as and when specific problems are detected. To enable the detailed diagnosis capability of this test for a particular server, choose the On option. To disable the capability, click on the Off option. The option to selectively enable/disable the detailed diagnosis capability will be available only if the following conditions are fulfilled: • The eG manager license should allow the detailed diagnosis capability • Both the normal and abnormal frequencies configured for the detailed diagnosis measures should not be 0.		
Outputs of the test	One set of results for the Dashboards Processing server running in the node monitored		
Measurements made by the	**Measurement**	**Measurement Unit**	**Interpretation**

<table>
<tr>
<td>test</td>
<td>Health state:

Indicates the current health state of the Dashboards Processing server in the monitored node.</td>
<td></td>
<td>The values that this measure can report and their corresponding numeric values are discussed in the table below:

<table><tr><th>Measure Value</th><th>Numeric Value</th></tr><tr><td>Danger</td><td>0</td></tr><tr><td>Caution</td><td>1</td></tr><tr><td>Healthy</td><td>2</td></tr></table>
If the server is not healthy, then the detailed diagnosis of this measure, if enabled, provides the process ID and processing plugin name of the server.

Note:

By default, this measure reports the Measure Values listed in the table above to indicate the health state of the server. In the graph of this measure however, the same is represented using the numeric values only.</td>
</tr>
<tr>
<td></td>
<td>Server running state:

Indicates the current running state of the Dashboards Processing server server.</td>
<td></td>
<td>The values that this measure can report and their corresponding numeric values are discussed in the table below:

<table><tr><th>Measure Value</th><th>Numeric Value</th></tr><tr><td>STOPPED</td><td>0</td></tr><tr><td>STARTING</td><td>1</td></tr><tr><td>INITIALIZING</td><td>2</td></tr><tr><td>RUNNING</td><td>3</td></tr><tr><td>STOPPING</td><td>4</td></tr><tr><td>FAILED</td><td>5</td></tr><tr><td>RUNNING_WITH_ERRORS</td><td>6</td></tr><tr><td>RUNNING_WITH_WARNINGS</td><td>7</td></tr></table>
Note:

By default, this measure reports the Measure Values listed in the table above to indicate the running state of the server. In the graph of this measure however, the same is represented using the numeric values only.</td>
</tr>
</table>

<table>
<tr>
<td></td>
<td>Server enabled state:
Indicates whether/not the Dashboards Processing server is enabled.</td>
<td></td>
<td>The values that this measure can report and their corresponding numeric values are discussed in the table below:
<table><tr><th>Measure Value</th><th>Numeric Value</th></tr><tr><td>Disabled</td><td>0</td></tr><tr><td>Enabled</td><td>1</td></tr></table>
Note:
By default, this measure reports the Measure Values listed in the table above to indicate whether/not the server is enabled. In the graph of this measure however, the same is represented using the numeric values only.</td>
</tr>
<tr>
<td></td>
<td>Busy server threads:
Indicates the number of server threads that are currently servicing requests.</td>
<td>Number</td>
<td>If this measure reaches the configured maximum thread pool size for the server, new requests to the server would have to wait until a server thread becomes free. If this happens often, it may significantly slowdown request processing by the server. In such a situation, you may want to consider resizing the thread pool.</td>
</tr>
</table>

1.3.10 Dashboards Server Performance Test

Users often expect the Dashboards Processing Server to accept all the dashboard requests they receive, process the requests in record time, and return EPF pages to them quickly, so that they can access just the pages of interest to them on demand, without waiting too long for reports that provide more information than they require. If a Dashboards Processing Server fails to live up to this expectation – i.e., if requests to a Dashboards Processing Server keep failing or if the server takes hours to process the requests it receives - user confidence in that server and in the SAP BOBI platform services in general will be destroyed. Also, constant request failures and consistent processing slowdowns on the server will also increase the length of the pending request queues on the server, choking that server eventually. In summary, processing bottlenecks with the Dashboards Processing Server can prove to be detrimental not only to the health of the server but also to the user experience with the SAP BOBI platform. This is why, it is imperative that administrators spot a probable slowdown or outage of the server early in its life cycle, so that they can avert such fatalities well before they occur. The **Dashboards Server Performance** test helps administrators with this.

This test closely tracks dashboard requests to the server, observes the rate at which the server processes these requests, rapidly isolates probable failures/slowdowns, and promptly alerts administrators to them. This way, the test provides administrators with a heads-up on potentially serious processing bottlenecks on the Dashboards Processing server, so that administrators can work towards preventing them.

Purpose	Closely tracks dashboard requests to the server, observes the rate at which the server processes these requests, rapidly isolates probable failures/slowdowns, and promptly alerts administrators to them. This way, the test provides administrators with a heads-up on potentially serious processing bottlenecks on the Dashboards Processing server, so that administrators can work

<table>
<tr><td></td><td colspan="3">towards preventing them.</td></tr>
<tr><td>Target of the test</td><td colspan="3">A SAP BOBI node</td></tr>
<tr><td>Agent deploying the test</td><td colspan="3">An internal/remote agent</td></tr>
<tr><td>Configurable parameters for the test</td><td colspan="3">1. TEST PERIOD - How often should the test be executed
2. HOST - Host name of the server for which the test is to be configured
3. PORT - Enter the port to which the specified HOST listens. This should be the port at which the web application server hosting SAP BOBI listens.
4. JMX REMOTE PORT - Specify the RMI port number of the BOBI monitoring application. To know the RMI port number of the monitoring application, refer to Section 1.1.1 of this document.
5. JNDI NAME - Specify the lookup name for connecting to the JMX connector of the BOBI monitoring application. To know the JNDI name, refer to Section 1.1.1 of this document.
6. JMX USER and JMX PASSWORD – Enter the credentials of an enterprise authenticated BOBI user belonging to the default monitoring users group.
7. CONFIRM PASSWORD - Confirm the password by retyping it here.
8. NODE NAME – Specify the name of the BOBI node being monitored.</td></tr>
<tr><td>Outputs of the test</td><td colspan="3">One set of results for the Dashboards Processing server running in the node monitored</td></tr>
<tr><td rowspan="3">Measurements made by the test</td><td>Measurement</td><td>Measurement Unit</td><td>Interpretation</td></tr>
<tr><td>Requests served rate:
Indicates the rate at which the server services the dashboard requests it receives.</td><td>Requests/Sec</td><td>A consistent drop in the value of this measure could indicate a probable processing slowdown.</td></tr>
<tr><td>Number of open jobs:
Indicates the number of jobs that the server and its child processes are currently processing.</td><td>Number</td><td>This is a good indicator of the current workload of the server.</td></tr>
<tr><td></td><td>Average processing time:
Indicates the average time taken by the server to process a request (taken over the last 500 requests).</td><td>Millisecs</td><td>Ideally, the value of this measure should be low. A high value or a consistent rise in this value is a cause for concern, as it indicates a slowdown in request processing.</td></tr>
</table>

	Maximum processing time: Indicates the maximum time spent by the server for processing a request (taken over the last 500 requests).	Number	
	Minimum processing time: Indicates the miimum time spent by the server for processing a request (taken over the last 500 requests).	Number	
	Number of queued requests: Indicates the number of requests waiting to be processed or are being processed.	Number	A steady rise in the value of this measure is a sign of a request queue that is growing. This means that the server is unable to process page requests as quickly as it receives them. You may want to check the value of the *Busy server threads* measure of the **Dashboards Processing Server Status** test to know if the lack of free threads in the thread pool is the reason for the slowness.
	Number of open connections: Indicates the number of connections open between the server and its clients.	Number	
	Percentage requests failed: Indicates what percentage of the last 500 requests have failed.	Percent	Ideally, the value of this measure should be 0. A non-zero value is a sign of trouble. A value close to 100 could hint at a critical problem condition that warrants an immediate and thorough investigation.
	Data transfer rate: Indicates the rate at which data is transferred from the server to its clients.	KB/Sec	

1.3.11 Report Application Server Test

The Report Application Server provides ad-hoc reporting capabilities that allow users to create and modify Crystal reports via the SAP Crystal Reports Server Embedded Software Development Kit (SDK). Users will not be able to customize existing reports or create new ones without this server process. Also, if this server is slow, user experience with reports will suffer terribly, destroying user confidence in the SAP BOBI platform. If such adversities are to be avoided, then administrators should be able to capture the failure or slowness of the Report Application server on-the-fly, investigate the reasons for the problem rapidly, and resolve them before end users notice and complain. The **Report Application Server** test makes this possible! This test keeps track of the health, status, and request processing ability of the Report Application server, and raises an alarm if the server suffers an outage or a slowdown, thus

enabling administrators to initiate preventive measures.

Purpose	keeps track of the health, status, and request processing ability of the Report Application server, and raises an alarm if the server suffers an outage or a slowdown, thus enabling administrators to initiate preventive measures		
Target of the test	A SAP BOBI node		
Agent deploying the test	An internal/remote agent		
Configurable parameters for the test	1. **TEST PERIOD** - How often should the test be executed 2. **HOST** - Host name of the server for which the test is to be configured 3. **PORT** - Enter the port to which the specified **HOST** listens. This should be the port at which the web application server hosting SAP BOBI listens. 4. **JMX REMOTE PORT** - Specify the RMI port number of the BOBI monitoring application. To know the RMI port number of the monitoring application, refer to Section 1.1.1 of this document. 5. **JNDI NAME** - Specify the lookup name for connecting to the JMX connector of the BOBI monitoring application. To know the JNDI name, refer to Section 1.1.1 of this document. 6. **JMX USER** and **JMX PASSWORD** – Enter the credentials of an enterprise authenticated BOBI user belonging to the default *monitoring users* group. 7. **CONFIRM PASSWORD** - Confirm the password by retyping it here. 8. **NODE NAME** – Specify the name of the BOBI node being monitored. 9. **DETAILED DIAGNOSIS** - To make diagnosis more efficient and accurate, the eG Enterprise suite embeds an optional detailed diagnostic capability. With this capability, the eG agents can be configured to run detailed, more elaborate tests as and when specific problems are detected. To enable the detailed diagnosis capability of this test for a particular server, choose the On option. To disable the capability, click on the Off option. The option to selectively enable/disable the detailed diagnosis capability will be available only if the following conditions are fulfilled: • The eG manager license should allow the detailed diagnosis capability • Both the normal and abnormal frequencies configured for the detailed diagnosis measures should not be 0.		
Outputs of the test	One set of results for the Report Application server running in the node monitored		
Measurements made by the	**Measurement**	**Measurement Unit**	**Interpretation**

<table>
<tr>
<td>test</td>
<td>Health state:

Indicates the current health state of the Report Application server in the monitored node.</td>
<td></td>
<td>The values that this measure can report and their corresponding numeric values are discussed in the table below:

<table><tr><th>Measure Value</th><th>Numeric Value</th></tr><tr><td>Danger</td><td>0</td></tr><tr><td>Caution</td><td>1</td></tr><tr><td>Healthy</td><td>2</td></tr></table>
If the server is not healthy, then the detailed diagnosis of this measure, if enabled, provides the process ID of the server.

Note:

By default, this measure reports the Measure Values listed in the table above to indicate the health state of the server. In the graph of this measure however, the same is represented using the numeric values only.</td>
</tr>
<tr>
<td></td>
<td>Server running state:

Indicates the current running state of the Report Application server.</td>
<td></td>
<td>The values that this measure can report and their corresponding numeric values are discussed in the table below:

<table><tr><th>Measure Value</th><th>Numeric Value</th></tr><tr><td>STOPPED</td><td>0</td></tr><tr><td>STARTING</td><td>1</td></tr><tr><td>INITIALIZING</td><td>2</td></tr><tr><td>RUNNING</td><td>3</td></tr><tr><td>STOPPING</td><td>4</td></tr><tr><td>FAILED</td><td>5</td></tr><tr><td>RUNNING_WITH_ERRORS</td><td>6</td></tr><tr><td>RUNNING_WITH_WARNINGS</td><td>7</td></tr></table>
Note:

By default, this measure reports the Measure Values listed in the table above to indicate the running state of the server. In the graph of this measure however, the same is represented using the numeric values only.</td>
</tr>
</table>

	Server enabled state: Indicates whether/not the Report Application server is enabled.		The values that this measure can report and their corresponding numeric values are discussed in the table below: **Measure Value** / **Numeric Value** Disabled / 0 Enabled / 1 **Note:** By default, this measure reports the **Measure Value**s listed in the table above to indicate whether/not the server is enabled. In the graph of this measure however, the same is represented using the numeric values only.
	Busy server threads: Indicates the number of server threads that are currently servicing requests.	Number	If this measure reaches the configured maximum thread pool size for the server, new requests to the server would have to wait until a server thread becomes free. If this happens often, it may significantly slowdown request processing by the server. In such a situation, you may want to consider resizing the thread pool.

1.3.12 Web Intelligence Server Status Test

SAP BusinessObjects Web Intelligence ensures your deployment meets performance demands and supports standardization efforts by providing an integrated analysis tool for all users. It is a tool built upon a complete, trusted, and agile BI platform that offers powerful, online and offline ad hoc query and reporting.

If the Web Intelligence server is not running or is experiencing errors or is disabled, then critical data cannot be queried from storage and custom reports cannot be built. Likewise, if the Web Intelligence server is not sized with adequate threads, slowdowns in query processing become inevitable. To avoid this, administrators must be promptly alerted when a Web Intelligence server suddenly stops functioning or is about to run out of free threads. This is exactly what the **Web Intelligence Server Status** test does. This test tracks the overall health and status of the Web Intelligence server and notifies administrators if the server has stopped, has encountered errors, or is disabled. In addition, the test also monitors the thread pool usage of the server and proactively alerts administrators if the pool is over-utilized.

Purpose	Tracks the overall health and status of the Web Intelligence server and notifies administrators if the server has stopped, has encountered errors, or is disabled. In addition, the test also monitors the thread pool usage of the server and proactively alerts administrators if the pool is over-utilized.
Target of the test	A SAP BOBI node
Agent deploying the test	An internal/remote agent

<table>
<tr><td>Configurable parameters for the test</td><td colspan="3">1. TEST PERIOD - How often should the test be executed
2. HOST - Host name of the server for which the test is to be configured
3. PORT - Enter the port to which the specified HOST listens. This should be the port at which the web application server hosting SAP BOBI listens.
4. JMX REMOTE PORT - Specify the RMI port number of the BOBI monitoring application. To know the RMI port number of the monitoring application, refer to Section 1.1.1 of this document.
5. JNDI NAME - Specify the lookup name for connecting to the JMX connector of the BOBI monitoring application. To know the JNDI name, refer to Section 1.1.1 of this document.
6. JMX USER and JMX PASSWORD – Enter the credentials of an enterprise authenticated BOBI user belonging to the default monitoring users group.
7. CONFIRM PASSWORD - Confirm the password by retyping it here.
8. NODE NAME – Specify the name of the BOBI node being monitored.
9. DETAILED DIAGNOSIS - To make diagnosis more efficient and accurate, the eG Enterprise suite embeds an optional detailed diagnostic capability. With this capability, the eG agents can be configured to run detailed, more elaborate tests as and when specific problems are detected. To enable the detailed diagnosis capability of this test for a particular server, choose the On option. To disable the capability, click on the Off option.
The option to selectively enable/disable the detailed diagnosis capability will be available only if the following conditions are fulfilled:
• The eG manager license should allow the detailed diagnosis capability
• Both the normal and abnormal frequencies configured for the detailed diagnosis measures should not be 0.</td></tr>
<tr><td>Outputs of the test</td><td colspan="3">One set of results for Web Intelligence server running in the node monitored</td></tr>
<tr><td>Measurements made by the test</td><td>Measurement</td><td>Measurement Unit</td><td>Interpretation</td></tr>
</table>

<table>
<tr>
<td></td>
<td>Health state:

Indicates the current health state of the Web Intelligence server in the monitored node.</td>
<td></td>
<td>The values that this measure can report and their corresponding numeric values are discussed in the table below:

<table>
<tr><th>Measure Value</th><th>Numeric Value</th></tr>
<tr><td>Danger</td><td>0</td></tr>
<tr><td>Caution</td><td>1</td></tr>
<tr><td>Healthy</td><td>2</td></tr>
</table>

If the server is not healthy, the detailed diagnosis of this measure (if enabled) provides the process ID of the server

Note:

By default, this measure reports the Measure Values listed in the table above to indicate the health state of the server. In the graph of this measure however, the same is represented using the numeric values only.</td>
</tr>
<tr>
<td></td>
<td>Server running state:

Indicates the current running state of the Web Intelligence server.</td>
<td></td>
<td>The values that this measure can report and their corresponding numeric values are discussed in the table below:

<table>
<tr><th>Measure Value</th><th>Numeric Value</th></tr>
<tr><td>STOPPED</td><td>0</td></tr>
<tr><td>STARTING</td><td>1</td></tr>
<tr><td>INITIALIZING</td><td>2</td></tr>
<tr><td>RUNNING</td><td>3</td></tr>
<tr><td>STOPPING</td><td>4</td></tr>
<tr><td>FAILED</td><td>5</td></tr>
<tr><td>RUNNING_WITH_ERRORS</td><td>6</td></tr>
<tr><td>RUNNING_WITH_WARNINGS</td><td>7</td></tr>
</table>

Note:

By default, this measure reports the Measure Values listed in the table above to indicate the running state of the server. In the graph of this measure however, the same is represented using the numeric values only.</td>
</tr>
</table>

	Server enabled state: Indicates whether/not the Web Intelligence server is enabled.		The values that this measure can report and their corresponding numeric values are discussed in the table below: **Measure Value** / **Numeric Value** Disabled / 0 Enabled / 1 **Note:** By default, this measure reports the **Measure Value**s listed in the table above to indicate whether/not the server is enabled. In the graph of this measure however, the same is represented using the numeric values only.
	Busy server threads: Indicates the number of server threads that are currently servicing requests.	Number	If this measure reaches the configured maximum thread pool size for the server, new requests to the server would have to wait until a server thread becomes free. If this happens often, it may significantly slowdown request processing by the server. In such a situation, you may want to consider resizing the thread pool.

1.3.13 Web Intelligence Server Cache Test

The Web Intelligence Server Cache significantly improves load time of documents by servicing requests using documents that have already been viewed. However, if this cache is sized poorly, it will not be able to hold many documents, forcing requests to be routed to the Web Intelligence Server for processing; this is bound to increase processing overheads and consequently, the load time of documents. To avoid this, administrators should track cache usage and tune the cache size based on how it is used. This is what the **Web Intelligence Server Cache** test helps administrators do.

This test tracks the load on the cache vis-à-vis its size and notifies administrators if the cache does not have enough memory resources to cater to the demand for documents. Additionally, the test also provides effective pointers to how cache size can be increased and usage can be optimized.

Purpose	Tracks the load on the cache vis-à-vis its size and notifies administrators if the cache does not have enough memory resources to cater to the demand for documents. Additionally, the test also provides effective pointers to how cache size can be increased and usage can be optimized.
Target of the test	A SAP BOBI node
Agent deploying the test	An internal/remote agent

<table>
<tr><td>Configurable parameters for the test</td><td colspan="3">1. TEST PERIOD - How often should the test be executed
2. HOST - Host name of the server for which the test is to be configured
3. PORT - Enter the port to which the specified HOST listens. This should be the port at which the web application server hosting SAP BOBI listens.
4. JMX REMOTE PORT - Specify the RMI port number of the BOBI monitoring application. To know the RMI port number of the monitoring application, refer to Section 1.1.1 of this document.
5. JNDI NAME - Specify the lookup name for connecting to the JMX connector of the BOBI monitoring application. To know the JNDI name, refer to Section 1.1.1 of this document.
6. JMX USER and JMX PASSWORD – Enter the credentials of an enterprise authenticated BOBI user belonging to the default monitoring users group.
7. CONFIRM PASSWORD - Confirm the password by retyping it here.
8. NODE NAME – Specify the name of the BOBI node being monitored.</td></tr>
<tr><td>Outputs of the test</td><td colspan="3">One set of results for the Web Intelligence Server in the node monitored</td></tr>
<tr><td rowspan="5">Measurements made by the test</td><td>Measurement</td><td>Measurement Unit</td><td>Interpretation</td></tr>
<tr><td>Cache used size:
Indicates the amount of data stored in the cache.</td><td>KB</td><td></td></tr>
<tr><td>Out-of-date document deletion rate:
Indicates the rate at which obsolete documents are deleted from the cache.</td><td>Docs/Sec</td><td>A higher rate is generally desired as it releases cache memory for use by new documents, and thus improves the cache's request handling ability.</td></tr>
<tr><td>Max cache reached rate:
Indicates the rate at which the server cache size reached the maximum size limit allowed on the server.</td><td></td><td>A high rate indicates that the cache is getting filled up very often. This implies that the cache may require a resizing, as the load on the server is high.</td></tr>
<tr><td>Current number of documents opened from cache:
Indicates the number of documents for which the last request result has been directly read from the cache.</td><td>Number</td><td rowspan="2">By comparing these two measures at a given point in time, you can measure how effectively the cache has been servicing document requests to the Web Intelligence server. If this comparison reveals ineffective cache usage, then you may want to check the cache size to see if the poor size of the cache is responsible for this.</td></tr>
<tr><td>Total documents:
Indicates the total number of documents that are currently open on the server.</td><td>Number</td></tr>
</table>

	Documents opened from cache: Indicates the percentage of documents opened from the cache.	Percent	Ideally, the value of this measure should be high. A low value is indicative of inoptimal cache usage. You may want to check the cache size in this case to see if the poor size of the cache is responsible for this.
	Number of documents with swap requests: Indicates the number of documents for which a clean up thread has scheduled swap requests.	Number	
	Documents swapped: Indicates the percentage of documents that have been swapped by swap requests.	Percent	Swapping ensures that inactive documents in the cache are swapped to the hard disk, so that the memory they have been hogging up until then can be released for use of subsequent documents. A high percentage of documents swapped is good news, as it increases unused cache memory and enables the cache to hold more documents, which translates into more requests served.

1.3.14 Web Intelligence Server Performance Test

How well the Web Intelligence Server processes requests depends upon many factors, including:

- The amount of CPU resources allocated to the server;
- The amount of memory allocated to the server:
- The thread pool size

If the server experiences CPU/memory contentions or runs short of free threads, it is bound to affect the speed with which it processes requests, thus resulting in a server slowdown. This is why, administrators are advised to periodically run the **Web Intelligence Server Performance** test. At pre-configured intervals, this test measures the CPU and memory usage of the Web Intelligence server and alerts administrators to threshold violations (if any). In addition, the test also checks for free threads in the thread pool, and if only a few free threads are available, warns administrators of a potential processing bottleneck.

Purpose	Measures the CPU and memory usage of the Web Intelligence server and alerts administrators to threshold violations (if any). In addition, the test also checks for free threads in the thread pool, and if only a few free threads are available, warns administrators of a potential processing bottleneck.
Target of the test	A SAP BOBI node
Agent deploying the test	An internal/remote agent

<table>
<tr><td>Configurable parameters for the test</td><td colspan="3">1. TEST PERIOD - How often should the test be executed
2. HOST - Host name of the server for which the test is to be configured
3. PORT - Enter the port to which the specified HOST listens. This should be the port at which the web application server hosting SAP BOBI listens.
4. JMX REMOTE PORT - Specify the RMI port number of the BOBI monitoring application. To know the RMI port number of the monitoring application, refer to Section 1.1.1 of this document.
5. JNDI NAME - Specify the lookup name for connecting to the JMX connector of the BOBI monitoring application. To know the JNDI name, refer to Section 1.1.1 of this document.
6. JMX USER and JMX PASSWORD – Enter the credentials of an enterprise authenticated BOBI user belonging to the default monitoring users group.
7. CONFIRM PASSWORD - Confirm the password by retyping it here.
8. NODE NAME – Specify the name of the BOBI node being monitored.</td></tr>
<tr><td>Outputs of the test</td><td colspan="3">One set of results for the Web Intelligence Server in the node monitored</td></tr>
<tr><td rowspan="3">Measurements made by the test</td><td>Measurement</td><td>Measurement Unit</td><td>Interpretation</td></tr>
<tr><td>CPU usage:
Indicates the percentage of CPU resources used since the server started.</td><td>Percent</td><td>A value close to 100% is indicative of a CPU contention on the server.</td></tr>
<tr><td>Memory high threshold violation rate:
Indicates the rate at which the the Memory Upper Threshold was violated by the server in the last measurement period.</td><td>Violations/Sec</td><td>If Memory Analysis property is enabled on the server, then the following properties can be configured and tracked on the server:
• Memory Maximum Threshold
• Memory Upper Threshold</td></tr>
</table>

	Memory max threshold violation rate: Indicates the rate at which the the *Memory Maximum Threshold* was violated by the server in the last measurement period.	Violations/Sec	• Memory Lower Threshold When the server's process memory is above the *Memory Upper Threshold*, the only operation that is allowed is saving documents. When the process memory is above the *Memory Maximum Threshold*, all operations stop and fail. It is evident then that such threshold violations are not good news! Ideally therefore, the value of the *Memory high threshold violation rate* and the *Memory max threshold violation rate* measures should be 0. A non-zero value is a sign of frequent memory threshold violations, which could either indicate a serious memory crunch on the server or an improper threshold setting. If further investigations reveal a memory contention on the server, make sure that you allocate more memory to the server to avoid the same.
	Virtual memory size: Indicates the amount of memory assigned to the server.	MB	
	Active threads: Indicates the number of threads (from the asynchronous thread pool) that are currently serving user requests to the server.	Number	If this measure reaches the configured maximum thread pool size for the server, new requests to the server would have to wait until a server thread becomes free. If this happens often, it may significantly slowdown request processing by the server. In such a situation, you may want to consider resizing the thread pool.

1.3.15 Web Intelligence Server Activity Test

By observing the level of activity on the Web Intelligence Server over time and by monitoring how the server handles the workload generated by these activities, administrators can quickly figure out whether/not the server possesses the processing power required to service the load. Such useful insights on workload and throughput is provided by the **Web Intelligence Server Activity** test. This test closely tracks the clients calls made to and sessions created on the Web Intelligence server, reports the rate at which the server executes tasks in response to these calls, and thus indicates how quickly/slowly the server is executing these tasks. Processing bottlenecks on the server can be identified in the process.

Purpose	Closely tracks the clients calls made to and sessions created on the Web Intelligence server, reports the rate at which the server executes tasks in response to these calls, and thus indicates how quickly/slowly the server is executing these tasks. Processing bottlenecks on the server can be identified in the process.
Target of the	A SAP BOBI node

<table>
<tr><td>test</td><td colspan="3"></td></tr>
<tr><td>Agent deploying the test</td><td colspan="3">An internal/remote agent</td></tr>
<tr><td>Configurable parameters for the test</td><td colspan="3">1. TEST PERIOD - How often should the test be executed
2. HOST - Host name of the server for which the test is to be configured
3. PORT - Enter the port to which the specified HOST listens. This should be the port at which the web application server hosting SAP BOBI listens.
4. JMX REMOTE PORT - Specify the RMI port number of the BOBI monitoring application. To know the RMI port number of the monitoring application, refer to Section 1.1.1 of this document.
5. JNDI NAME - Specify the lookup name for connecting to the JMX connector of the BOBI monitoring application. To know the JNDI name, refer to Section 1.1.1 of this document.
6. JMX USER and JMX PASSWORD – Enter the credentials of an enterprise authenticated BOBI user belonging to the default monitoring users group.
7. CONFIRM PASSWORD - Confirm the password by retyping it here.
8. NODE NAME – Specify the name of the BOBI node being monitored.</td></tr>
<tr><td>Outputs of the test</td><td colspan="3">One set of measures for the Web Intelligence Server on the monitored node</td></tr>
<tr><td rowspan="4">Measurements made by the test</td><td>Measurement</td><td>Measurement Unit</td><td>Interpretation</td></tr>
<tr><td>Current number of client calls:
Indicates the current number of CORBA calls that the server is processing.</td><td>Number</td><td>This is a good indicator of the level of client activity on the server.</td></tr>
<tr><td>Client call rate:
Indicates the rate at which the CORBA calls were received by the server in the last measure period.</td><td>Calls/Sec</td><td>A consistent increase in the value of this measure is indicative of a steady rise in server workload.</td></tr>
<tr><td>Current number of tasks executed:
Indicates the number of tasks that are currently executed by the server.</td><td>Number</td><td></td></tr>
<tr><td></td><td>Task execution rate:
Indicates the rate at which the server executes tasks.</td><td>Executions/Sec</td><td>A steady drop in the value of this measure is a cause for concern as it indicates that the server is unable to process tasks as quickly as it receives client calls/requests. This is a sign of a processing bottleneck on the server and requires further investigation.</td></tr>
</table>

	Active sessions: Indicates the number of sessions that are currently able to accept requests from clients.	Number	This is a good indicator of the current session load on the server.
	Created sessions: Indicates the number of sessions that are currently created on the server.	Number	
	Session creation rate: Indicates the rate at which sessions were created.	Sessions/Sec	
	Session timeout rate: Indicates the rate at which sessions timed out.	Sessions/Sec	A high timeout rate indicates that sessions are getting timed out very often. You may want to consider changing the Timeout value for sessions to reduce the frequent session logouts.
	Number of users: Indicates the number of users currently connected to the server.	Number	This is a good indicator of the user load on the server.

1.3.16 Process Logs Test

Logs related to the core server types – namely, Central Management server, Adaptive Processing server and Adaptive Job server - are periodically checked by this test, so that critical problem events can be quickly captured and resolved.

Purpose	Logs related to the core server types – namely, Central Management server, Adaptive Processing server and Adaptive Job server - are periodically checked by this test, so that critical problem events can be quickly captured and resolved.
Target of the test	A SAP BOBI node
Agent deploying the test	An internal/remote agent

Configurable parameters for the test	1. **TEST PERIOD** - How often should the test be executed 2. **HOST** - Host name of the server for which the test is to be configured 3. **PORTNO** - Enter the port to which the specified **HOST** listens 4. **LOG DIRECTORY** - This is the directory to which logs from various nodes installed on a host are written. Typically, these logs are written to '*.glf' files in the 'logging' directory of the BOBI installation. If SAP BOBI is installed in the C drive of a Windows host, the *logging* directory will be available in the following location by default: *C:|Program Files (x86)|SAP BusinessObjects|SAP BusinessObjects Enterprise XI 4.0|* 5. **SERVER ABBREVIATIONS** - Log file names are generally of the following fomat : *<server abbreviation>_<node name>.<server type>*.glf'* For e.g., *cms_SAPBOBI.CentralManagementServer_trace.001284* is one of the log trace files from the Central Management Server running in the node called SAPBOBI. Server abbreviation in this case is **cms**. The default value for this parameter has hence been set as a comma separated list of server descriptions and their abbreviations as follows *: <server description>:<server abbreviation>*. For trace files of the Central Management Server, Adaptive Processing Server and Adaptive Job Server, this parameter has been by default set as: *Central Management Servers:CMS,Adaptive Processing Servers:aps,Adaptive Job Servers:jobserver*. 6. **SEARCHPATTERN** - Enter the specific patterns of messages to be monitored. The pattern should be in the following format: *<PatternName>:<Pattern>*, where *<PatternName>* is the pattern name that will be displayed in the monitor interface and *<Pattern>* is an expression of the form - *expr* or expr or *expr or expr*, etc. A leading '*' signifies any number of leading characters, while a trailing '*' signifies any number of trailing characters. For example, say you specify ORA:ORA-* in the **SEARCHPATTERN** text box. This indicates that "ORA" is the pattern name to be displayed in the monitor interface. "ORA-*" indicates that the test will monitor only those lines in the log file which start with the term "ORA-". Similarly, if your pattern specification reads: offline:*offline, then it means that the pattern name is offline and that the test will monitor those lines in the log file which end with the term offline. A single pattern may also be of the form e1+e2, where + signifies an OR condition. That is, the *<PatternName>* is matched if either e1 is true or e2 is true. Multiple search patterns can be specified as a comma-separated list. For example: ORA:ORA-*,offline:*offline*,online:*online. Each of these patterns will be searched for in every log file that is present in the configured **LOG DIRECTORY**. 7. **LINES** - Specify two numbers in the format x:y. This means that when a line in the log file matches a particular pattern, then x lines before the matched line and y lines after the matched line will be reported in the detail diagnosis output (in addition to the matched line). The default value here is 0:0. Multiple entries can be provided as a comma-separated list. If you give 1:1 as the value for **LINES**, then this value will be applied to all the patterns specified in the **SEARCHPATTERN** field. If you give 0:0,1:1,2:1 as the value for **LINES** and if the corresponding value in the **SEARCHPATTERN** field is like ORA:ORA-*,offline:*offline*,online:*online then: 0:0 will be applied to ORA:ORA-* pattern 1:1 will be applied to offline:*offline* pattern 2:1 will be applied to online:*online pattern

8. **EXCLUDEPATTERN** - Provide a comma-separated list of patterns to be excluded from monitoring in the **EXCLUDEPATTERN** text box. For example **critical*,*exception**. By default, this parameter is set to 'none'.

9. **UNIQUEMATCH** - By default, the **UNIQUEMATCH** parameter is set to **FALSE**, indicating that, by default, the test checks every line in the log file for the existence of each of the configured **SEARCHPATTERNS**. By setting this parameter to **TRUE**, you can instruct the test to ignore a line and move to the next as soon as a match for one of the configured patterns is found in that line. For example, assume that *Pattern1:*fatal*,Pattern2:*error** is the **SEARCHPATTERN** that has been configured. If **UNIQUEMATCH** is set to **FALSE**, then the test will read every line in the log file completely to check for the existence of messages embedding the strings 'fatal' and 'error'. If both the patterns are detected in the same line, then the number of matches will be incremented by 2. On the other hand, if **UNIQUEMATCH** is set to **TRUE**, then the test will read a line only until a match for one of the configured patterns is found and not both. This means that even if the strings 'fatal' and 'error' follow one another in the same line, the test will consider only the first match and not the next. The match count in this case will therefore be incremented by only 1.

10. **ROTATINGFILE** - This flag governs the display of descriptors for this test in the eG monitoring console.

    If this flag is set to **true**, the descriptors of this test will be displayed in the following format: *<Full_Path_to_LogDirectory>:<ServerAbbreviation>*. For instance, if the **LOG DIRECTORY** parameter is set to *c:|SAPBOBI|logs* and **ROTATINGFILE** is set to **true**, then, your descriptor will be of the following format: *c:|SAPBOBI|logs:<ServerAbbreviation>*. On the other hand, if the **ROTATINGFILE** flag had been set to **false**, then the descriptors will be of the following format: *<LogDirectory>:<ServerAbbreviation>* - i.e., *logs:<ServerAbbreviation>* in the case of the example above.

11. **USEUTF8** - If UTF-8 encoding is to be used for reading the log files in the configured **LOG DIRECTORY**, then, set the **USEUTF8** flag to **true**. By default, this flag is set to **false**.

12. **USEUTF16** - If UTF-16 encoding is to be used for reading the log files in the configured **LOG DIRECTORY**, then, set the **USEUTF16** flag to **true**. By default, this flag is set to **false**.

13. **ENCODEFORMAT** – By default, this is set to *none*, indicating that no encoding format applies by default. However, if the test has to use a specific encoding format for reading from the log files in the **LOG DIRECTORY**, then you will have to provide a valid encoding format here - eg., *UTF-8*.

14. **NODE NAME** – Specify the name of the BOBI node being monitored.

	15. **DD FREQUENCY** - Refers to the frequency with which detailed diagnosis measures are to be generated for this test. The default is *1:1*. This indicates that, by default, detailed measures will be generated every time this test runs, and also every time the test detects a problem. You can modify this frequency, if you so desire. Also, if you intend to disable the detailed diagnosis capability for this test, you can do so by specifying *none* against **DD FREQUENCY**. 16. **DETAILED DIAGNOSIS** - To make diagnosis more efficient and accurate, the eG Enterprise suite embeds an optional detailed diagnostic capability. With this capability, the eG agents can be configured to run detailed, more elaborate tests as and when specific problems are detected. To enable the detailed diagnosis capability of this test for a particular server, choose the **On** option. To disable the capability, click on the **Off** option. The option to selectively enable/disable the detailed diagnosis capability will be available only if the following conditions are fulfilled: • The eG manager license should allow the detailed diagnosis capability • Both the normal and abnormal frequencies configured for the detailed diagnosis measures should not be 0.		
Outputs of the test	One set of results for the **SERVER ABBREVIATION** configured		
Measurements made by the test	**Measurement**	**Measurement Unit**	**Interpretation**
	High importance messages: Indicates the number of log messages with importance set to 'high' in the last measure period.	Messages	High importance designation for the log message is characterized by the '>=' symbol. For each such message, detailed diagnosis provides the details such as location (message source), timestamp, trace category, server, message, user etc.
	Highest importance messages: Indicates the number of log messages with importance set as 'highest' in the last measure period.	Messages	Highest importance designation for the log message is characterized by the '>>' symbol. For each such message, detailed diagnosis provides the details such as location (message source), timestamp, trace category, server, message, user etc.,
	Errors: Indicates the number of error log messages in the last measure period.	Messages	Error log messages have Severity values set to 'E'. For each such message, detailed diagnosis provides the details such as Location (message source), timestamp, trace category, server, message, user etc.
	Asserts: Indicates the number of Assert log messages in the last measure period.	Messages	Assert log messages have Severity values set to 'A'. For each such message, detailed diagnosis provides the details such as location (message source), timestamp, trace category, server, message, user etc.,

	Fatal messages: Indicates the number of fatal log messages in the last measure period.	Messages	Fatal log messages have severity values set to 'F'. For each such message, detailed diagnosis provides the details such as location (message source), timestamp, trace category, server, message, user etc.,
	New messages: Indicates the number of log messages in the last measure period.	Messages	
	Problem messages rate: Indicates the rate of all problem messages in the last measure period.	Messages/Sec	A high value is a cause for concern as it indicates that problems are occurring frequently. Compare the value of this measure across **SERVER ABBREVIATIONS** to identify the server that is the most problem-prone.

1.3.17 Dimensional Semantic Layer Bridge Service Test

The semantic layer allows SAP BusinessObjects Web Intelligence to deliver documents, by utilizing multiple synchronized data providers, including online analytical processing (OLAP) and common warehousing metamodel (CWM) data sources. Tracking the connections and requests to the semantic layer will help administrators assess how efficiently the semantic layer handles the requests it receives and identify bottlenecks (if any). To monitor the load on the semantic layer and measure its load-processing ability, administrators can use the **Dimensional Semantic Layer Bridge Service** test. This test keeps an eye on the requests received by and connections to the semantic layer and reports the count of data requests still to be serviced by the layer, thus pointing to probable processing bottlenecks.

Purpose	Keeps an eye on the requests received by and connections to the semantic layer and reports the count of data requests still to be serviced by the layer, thus pointing to probable processing bottlenecks
Target of the test	A SAP BOBI node
Agent deploying the test	An internal/remote agent

Configurable parameters for the test	1. **TEST PERIOD** - How often should the test be executed 2. **HOST** - Host name of the server for which the test is to be configured 3. **PORT** - Enter the port to which the specified **HOST** listens. This should be the port at which the web application server hosting SAP BOBI listens. 4. **JMX REMOTE PORT** - Specify the RMI port number of the BOBI monitoring application. To know the RMI port number of the monitoring application, refer to Section 1.1.1 of this document. 5. **JNDI NAME** - Specify the lookup name for connecting to the JMX connector of the BOBI monitoring application. To know the JNDI name, refer to Section 1.1.1 of this document. 6. **JMX USER** and **JMX PASSWORD** – Enter the credentials of an enterprise authenticated BOBI user belonging to the default *monitoring users* group. 7. **CONFIRM PASSWORD** - Confirm the password by retyping it here. 8. **NODE NAME** – Specify the name of the BOBI node being monitored.		
Outputs of the test	One set of results for the Adaptive Processing Server in the node monitored		
Measurements made by the test	**Measurement**	**Measurement Unit**	**Interpretation**
	Connections: Indicates the number of connections currently open on the semantic layer.	Number	This is a good indicator of the current connection load on the layer.
	OLAP connections: Indicates the number of OLAP connections currently open on the semantic layer.	Number	This is a good indicator of the current OLAP connection load on the layer.
	Sessions: Indicates the number of open sessions between clients and the layer.	Number	This is a good indicator of the current session load on the layer.
	Data requests: Indicates the number of currently open data requests between clients and the layer.	Number	A consistent increase in the value of this measure could indicate that requests are not been processed quickly enough by the server. You may have to investigate the reasons for this bottleneck and fix it.

1.4 The BOBI Management Layer

This layer monitors the status, workload, and performance of the Central Management Server running in the monitored node.

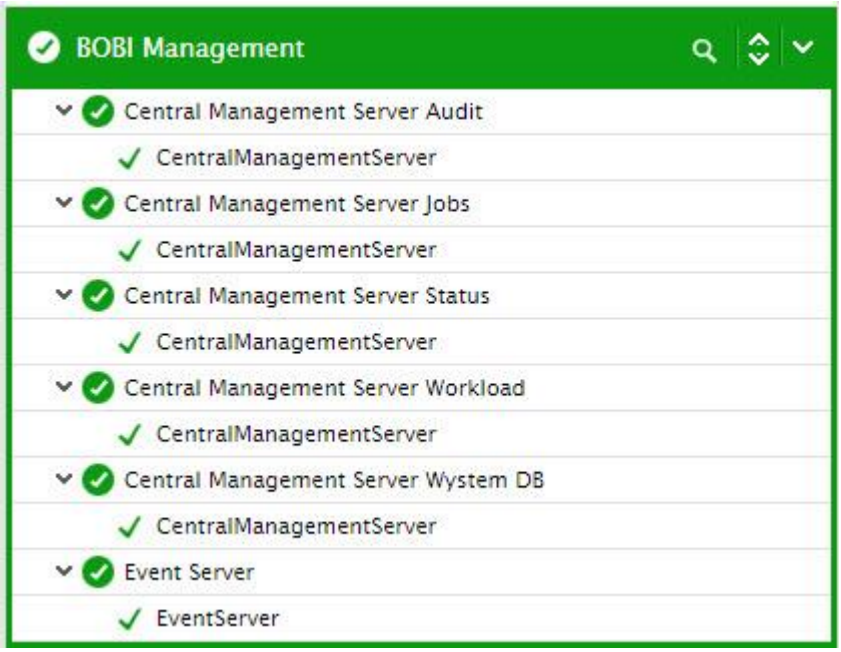

Figure 1. 10: The tests mapped to the BOBI Management layer

1.4.1 Central Management Server Status Test

The Central Management Server (CMS) provides and maintains security and configuration information, sends service requests to servers, manages auditing, and maintains the CMS system database. In the event of the failure of the CMS therefore, the operations of all components that make up the SAP BOBI platform can no longer be controlled or co-ordinated, resulting in total chaos! To ensure that the CMS is up and running 24 x 7 and provides uninterrupted security, request routing, and auditing services to the SAP BOBI components, administrators must run the **Central Management Server Status** test at frequent intervals. This test continuously monitors the availability and processing ability of the CMS and proactively alerts administrators of the probable slowdown/outage of the CMS. In the process, the test enables administrators to swiftly initiate preventive action, so that the potential performance anomalies are kept at bay.

Purpose	Continuously monitors the availability and processing ability of the CMS and proactively alerts administrators of the probable slowdown/outage of the CMS. In the process, the test enables administrators to swiftly initiate preventive action, so that the potential performance anomalies are kept at bay.
Target of the test	A SAP BOBI node
Agent deploying the test	An internal/remote agent

Configurable parameters for the test	1. **TEST PERIOD** - How often should the test be executed 2. **HOST** - Host name of the server for which the test is to be configured 3. **PORT** - Enter the port to which the specified **HOST** listens. This should be the port at which the web application server hosting SAP BOBI listens. 4. **JMX REMOTE PORT** - Specify the RMI port number of the BOBI monitoring application. To know the RMI port number of the monitoring application, refer to Section 1.1.1 of this document. 5. **JNDI NAME** - Specify the lookup name for connecting to the JMX connector of the BOBI monitoring application. To know the JNDI name, refer to Section 1.1.1 of this document. 6. **JMX USER** and **JMX PASSWORD** – Enter the credentials of an enterprise authenticated BOBI user belonging to the default *monitoring users* group. 7. **CONFIRM PASSWORD** - Confirm the password by retyping it here. 8. **NODE NAME** – Specify the name of the BOBI node being monitored. 9. **DETAILED DIAGNOSIS** - To make diagnosis more efficient and accurate, the eG Enterprise suite embeds an optional detailed diagnostic capability. With this capability, the eG agents can be configured to run detailed, more elaborate tests as and when specific problems are detected. To enable the detailed diagnosis capability of this test for a particular server, choose the On option. To disable the capability, click on the Off option. The option to selectively enable/disable the detailed diagnosis capability will be available only if the following conditions are fulfilled: • The eG manager license should allow the detailed diagnosis capability • Both the normal and abnormal frequencies configured for the detailed diagnosis measures should not be 0.		
Outputs of the test	One set of results for the CMS running in the node monitored		
Measurements made by the	**Measurement**	**Measurement Unit**	**Interpretation**

<table>
<tr>
<td>test</td>
<td>Health state:

Indicates the current health state of the CMS in the monitored node.</td>
<td></td>
<td>The values that this measure can report and their corresponding numeric values are discussed in the table below:

<table>
<tr><th>Measure Value</th><th>Numeric Value</th></tr>
<tr><td>Danger</td><td>0</td></tr>
<tr><td>Caution</td><td>1</td></tr>
<tr><td>Healthy</td><td>2</td></tr>
</table>

Note:

By default, this measure reports the Measure Values listed in the table above to indicate the health state of the server. In the graph of this measure however, the same is represented using the numeric values only.</td>
</tr>
<tr>
<td></td>
<td>Server running state:

Indicates the current running state of the CMS.</td>
<td></td>
<td>The values that this measure can report and their corresponding numeric values are discussed in the table below:

<table>
<tr><th>Measure Value</th><th>Numeric Value</th></tr>
<tr><td>STOPPED</td><td>0</td></tr>
<tr><td>STARTING</td><td>1</td></tr>
<tr><td>INITIALIZING</td><td>2</td></tr>
<tr><td>RUNNING</td><td>3</td></tr>
<tr><td>STOPPING</td><td>4</td></tr>
<tr><td>FAILED</td><td>5</td></tr>
<tr><td>RUNNING_WITH_ERRORS</td><td>6</td></tr>
<tr><td>RUNNING_WITH_WARNINGS</td><td>7</td></tr>
</table>

Note:

By default, this measure reports the Measure Values listed in the table above to indicate the running state of the server. In the graph of this measure however, the same is represented using the numeric values only.</td>
</tr>
</table>

	Server enabled state: Indicates whether/not the CMS is enabled.		The values that this measure can report and their corresponding numeric values are discussed in the table below: **Measure Value** / **Numeric Value** Disabled / 0 Enabled / 1 **Note:** By default, this measure reports the **Measure Value**s listed in the table above to indicate whether/not the server is enabled. In the graph of this measure however, the same is represented using the numeric values only.
	Busy server threads: Indicates the number of server threads that are currently servicing requests.	Number	If this measure reaches the configured maximum thread pool size for the server, new requests to the server would have to wait until a server thread becomes free. If this happens often, it may significantly slowdown request processing by the server. In such a situation, you may want to consider resizing the thread pool.

1.4.2 Central Management Server Audit Test

Auditing allows you to keep a record of significant events on servers and applications, which helps give you a picture of what information is being accessed, how it's being accessed and changed, and who is performing these operations.

The Central Management Server (CMS) acts as the system auditor, while each server or application that triggers an auditable event acts as an auditee. When an audited event is triggered, the auditee will generate a record and store it in a local temporary file. At regular intervals, the CMS communicates with the auditees to request these records and writes the data to a database called the Auditing Data Store (ADS).

If the CMS is not able to connect to the ADS, then audit records cannot be written to the ADS. Without audit records, custom audit reports cannot be generated. In the absence of such reports, administrators cannot audit accesses to servers/applications or detect logins that are suspect; this could pose a serious security threat.

Also, if the CMS is not able to write events to the ADS as quickly as the auditee sends records to it, the auditing load on the CMS is bound to increase, resulting in auditing delays.

By quickly detecting connection issues with ADS and processing delays with CMS and rapidly fixing them, administrators can ensure that auditing operations function without a glitch. The **Central Management Server Audit** test helps with this.

This test periodically checks the CMS – ADS connection and reports breaks (if any). In addition, the test monitors how quickly CMS writes event records to the ADS and highlights bottlenecks in the writing process. With the help of the actionable information the test provides, administrators can accurately identify what is ailing the auditing function and how it can be optimized.

Purpose	Periodically checks the CMS – ADS connection and reports breaks (if any). In addition, the test monitors how quickly CMS writes event records to the ADS and highlights bottlenecks in the writing process. With the help of the actionable information the test provides, administrators can accurately identify what is ailing the auditing function and how it can be optimized		
Target of the test	A SAP BOBI node		
Agent deploying the test	An internal/remote agent		
Configurable parameters for the test	1. **TEST PERIOD** - How often should the test be executed 2. **HOST** - Host name of the server for which the test is to be configured 3. **PORT** - Enter the port to which the specified **HOST** listens. This should be the port at which the web application server hosting SAP BOBI listens. 4. **JMX REMOTE PORT** - Specify the RMI port number of the BOBI monitoring application. To know the RMI port number of the monitoring application, refer to Section 1.1.1 of this document. 5. **JNDI NAME** - Specify the lookup name for connecting to the JMX connector of the BOBI monitoring application. To know the JNDI name, refer to Section 1.1.1 of this document. 6. **JMX USER** and **JMX PASSWORD** – Enter the credentials of an enterprise authenticated BOBI user belonging to the default *monitoring users* group. 7. **CONFIRM PASSWORD** - Confirm the password by retyping it here. 8. **NODE NAME** – Specify the name of the BOBI node being monitored. 9. **DETAILED DIAGNOSIS** - To make diagnosis more efficient and accurate, the eG Enterprise suite embeds an optional detailed diagnostic capability. With this capability, the eG agents can be configured to run detailed, more elaborate tests as and when specific problems are detected. To enable the detailed diagnosis capability of this test for a particular server, choose the On option. To disable the capability, click on the Off option. The option to selectively enable/disable the detailed diagnosis capability will be available only if the following conditions are fulfilled: • The eG manager license should allow the detailed diagnosis capability • Both the normal and abnormal frequencies configured for the detailed diagnosis measures should not be 0.		
Outputs of the test	One set of results for the CMS running in the node monitored		
Measurements made by the	**Measurement**	**Measurement Unit**	**Interpretation**

<table>
<tr>
<td>test</td>
<td>Connection to auditing database is established ?:

Indicates whether/not the CMS has a healthy connection to the ADS.</td>
<td></td>
<td>The values that this measure can report and their corresponding numeric values are discussed in the table below:

<table>
<tr><th>Measure Value</th><th>Numeric Value</th></tr>
<tr><td>No</td><td>0</td></tr>
<tr><td>Yes</td><td>1</td></tr>
</table>

If this measure reports the value No, then use the detailed diagnosis of this measure to know the time that has elapsed since the last time the CMS wrote audit records in the ADS. This will enable administrators to figure out how long the ADS has remained inaccessible to the CMS. The connection name and user name will also be displayed as part of the detailed diagnostics.

Note:

By default, this measure reports the Measure Values listed in the table above to indicate whether/not the CMS is able to connect to the ADS. In the graph of this measure however, the same is represented using the numeric values only.</td>
</tr>
<tr>
<td></td>
<td>Auditing thread utilization:

Indicates the percentage of the polling cycle during which the CMS is collecting data from auditees.</td>
<td>Percent</td>
<td>If this value approaches 100%, it implies that the CMS is still receiving events as the next polling cycle is due to start. This will cause many events to remain pending on the CMS, resulting in significant delays in writing events to the ADS. To fix this, you may want to consider configuring the ADS to receive data at a higher rate or decreasing the number of auditing events.</td>
</tr>
</table>

<table>
<tr>
<td></td>
<td>Is auditor?:

Indicates whether/not the CMS is the auditor.</td>
<td></td>
<td>The values that this measure can report and their corresponding numeric values are discussed in the table below:

<table><tr><th>Measure Value</th><th>Numeric Value</th></tr><tr><td>No</td><td>0</td></tr><tr><td>Yes</td><td>1</td></tr></table>
Note:

By default, this measure reports the Measure Values listed in the table above to indicate whether/not the CMS is the auditor. In the graph of this measure however, the same is represented using the numeric values only.</td>
</tr>
<tr>
<td></td>
<td>Time consumed for auditing:

Indicates the time taken by the CMS to perform auditing.</td>
<td>Mins</td>
<td>Polling cycle for healthy systems should be less than 20 minutes. Cycles more than 120 minutes indicate a very busy system and this needs to be fixed by setting the auditing DB to receive events at a higher data rate or decreasing the number of events being audited.</td>
</tr>
<tr>
<td></td>
<td>Current number of auditing events queue:

Reports the number of auditing events that an Auditee has recorded, but which have not yet been retrieved by the CMS Auditor.</td>
<td>Number</td>
<td>If this number increases without bound, it could mean indicate that auditing has not been configured properly or that the system is heavily loaded and generating auditing events faster than the auditor can retrieve them. When stopping servers, it is advisable to disable them first and wait for auditing events to be fully retrieved and this queue becomes empty. Otherwise, they may be retrieved only when this server has been restarted and the CMS polls for them. Critical events may be missed out as a result.</td>
</tr>
</table>

1.4.3 Central Management Server Jobs Test

The Central Management Server (CMS) maintains a database of information about your Business Intelligence (BI) platform system (in the CMS system database) and audited user actions (in the Auditing Data Store). All platform services are managed by the CMS. The CMS also controls access to the system files where documents are stored, and information on users, user groups, security levels, and content. In a nutshell, the CMS performs all jobs critical to the management of the users and services of the SAP BOBI platform. Naturally therefore, the frequent failure of these critical jobs can impair the services, hamper user productivity, and ultimately kill user experience with the SAP BOBI platform. Also, if too many jobs are found waiting on the CMS owing to insufficient processing power, service delivery and user authentication may be significantly delayed, once again impacting user experience. To avoid such situations, administrators can periodically run the **Central Management Server Jobs** test to quickly capture job failures and waiting jobs, ascertain the reasons for these anomalies rapidly, and fine-tune the CMS configuration to eliminate them.

Purpose	Quickly captures job failures and waiting jobs, helps administrators ascertain the reasons for these anomalies rapidly, and provides pointers to how the CMS configuration can be fine-tuned so that these anomalies can be completely eliminated.		
Target of the test	A SAP BOBI node		
Agent deploying the test	An internal/remote agent		
Configurable parameters for the test	1. **TEST PERIOD** - How often should the test be executed 2. **HOST** - Host name of the server for which the test is to be configured 3. **PORT** - Enter the port to which the specified **HOST** listens. This should be the port at which the web application server hosting SAP BOBI listens. 4. **JMX REMOTE PORT** - Specify the RMI port number of the BOBI monitoring application. To know the RMI port number of the monitoring application, refer to Section 1.1.1 of this document. 5. **JNDI NAME** - Specify the lookup name for connecting to the JMX connector of the BOBI monitoring application. To know the JNDI name, refer to Section 1.1.1 of this document. 6. **JMX USER** and **JMX PASSWORD** – Enter the credentials of an enterprise authenticated BOBI user belonging to the default *monitoring users* group. 7. **CONFIRM PASSWORD** - Confirm the password by retyping it here. 8. **NODE NAME** – Specify the name of the BOBI node being monitored.		
Outputs of the test	One set of results for the CMS running in the node monitored		
Measurements made by the	**Measurement**	**Measurement Unit**	**Interpretation**

test	**Failed jobs:** Indicates the number of jobs that failed.	Number	Ideally, the value of this measure should be 0. A high value is indicative of frequent job failures, and warrants immediate investigation.
	Pending jobs: Indicates the number of jobs that are scheduled, but not ready, to run because the scheduled time or event has not arrived.	Number	A steady rise in the value of this measure is a cause for concern, as it indicates that many jobs are awaiting execution. Over time, these may choke the CMS. To avoid this, you may want to reconfigure schedules or event configurations to ensure jobs keep executing quickly.
	Running jobs: Indicates the number of jobs currently running on the CMS.	Number	This is a good indicator of the current workload of the CMS.
	Completed jobs: Indicates the number of completed jobs.	Number	
	Waiting jobs: Indicates the number of waiting jobs.	Number	Typically, waiting jobs are those that wait for resources to become available. A very high value for this measure could either indicate that one/more jobs currently running are hogging the available resources or that the CMS is not sized properly. In the case of the latter, you may want to allocate more resources to the CMS or increase the size of its thread pool.

1.4.4 Central Management Server System DB Test

The CMS uses the CMS system database to store SAP BusinessObjects Business Intelligence platform information, such as user, server, folder, document, configuration, and authentication details. It is sometimes referred to as the system repository.

The CMS is configured with the maximum number of connections it can establish with the CMS system database. For optimal performance, the CMS must establish all these connections. However, if all the established connections are used up, subsequent requests to the CMS that require access to the CMS system database will not be processed. The repercussions of this can range from a user being denied access to the SAP BOBI platform to a critical server configuration change not being updated in the database. Such failures can adversely impact SAP BOBI platform services. This is why, it would be good practice for administrators to run the **Central Management Server System DB** test at frequent intervals.

This test keeps track of the CMS system database connections established by the CMS and how these connections are utilized. In the process, the test promptly alerts administrators if almost all the established connections have been used up. This way, the test helps administrators understand which database parameters need to be tuned to ensure continuous database availability for servicing requests.

Purpose	Keeps track of the CMS system database connections established by the CMS and how these connections are utilized. In the process, the test promptly alerts administrators if almost all the established connections have been used up. This way, the test helps administrators understand which database parameters need to be tuned to ensure continuous database availability for servicing requests.		
Target of the test	A SAP BOBI node		
Agent deploying the test	An internal/remote agent		
Configurable parameters for the test	1. **TEST PERIOD** - How often should the test be executed 2. **HOST** - Host name of the server for which the test is to be configured 3. **PORT** - Enter the port to which the specified **HOST** listens. This should be the port at which the web application server hosting SAP BOBI listens. 4. **JMX REMOTE PORT** - Specify the RMI port number of the BOBI monitoring application. To know the RMI port number of the monitoring application, refer to Section 1.1.1 of this document. 5. **JNDI NAME** - Specify the lookup name for connecting to the JMX connector of the BOBI monitoring application. To know the JNDI name, refer to Section 1.1.1 of this document. 6. **JMX USER** and **JMX PASSWORD** – Enter the credentials of an enterprise authenticated BOBI user belonging to the default *monitoring users* group. 7. **CONFIRM PASSWORD** - Confirm the password by retyping it here. 8. **NODE NAME** – Specify the name of the BOBI node being monitored.		
Outputs of the test	One set of results for the CMS running in the node monitored		
Measurements made by the	**Measurement**	**Measurement Unit**	**Interpretation**

test	**Established system database connections:** Indicates the number of connections the CMS was able to establish with the system database.	Number	The CMS tries to establish all of the requested system database connections as mentioned in its server properties. Default value for this is 14. If the number of established connections is lesser than the configured connections, the CMS may not be performing optimally. If the value of this measure is 0, it could mean that the CMS system database is unavailable – probably owing to a hardware or software failure or a network proble. In this case, the CMS goes into the " Waiting for resources" state. If the SAP BusinessObjects Business Intelligence platform deployment has multiple CMSs, then subsequent requests from other servers are forwarded to any CMSs in the cluster that have an active connection to the system database. While a CMS is in the " Waiting for resources" state, any current requests that do not require database access continue to be processed, but requests that require access to the CMS database will fail.
	Established system database connections currently used: Indicates the number of established system database connections that the CMS is currently using.	Number	If all the system database connections are typically in use, this suggests a bottleneck. In this case, the allowed number of connections can be increased in the server property to improve CMS performance.
	Pending system database requests: Indicates the number of requests for the CMS system database that are waiting for an available connection.	Number	Consistently high values for this measure suggest that the configured number of connections for the CMS is insufficient and should be increased to improve CMS performance. CMS system database tuning also improves CMS performance.
	Objects in CMS system cache: Indicates the current number of objects in the CMS system database cache.	Number	The default value for maximum objects in cache is 100000. While a large number of objects in cache can improve database performance, it can also cause excessive consumption of space in cache. If the cache is not sized right, then this can degrade database performance in the long run.

	Objects in CMS system database: Indicates the current number of objects in the CMS system database.	Number	CMS DB size corresponds to the number of objects such as users or documents. 100 MB of disk space typically stores 30000 objects.

1.4.5 Central Management Server Workload Test

The workload of a CMS is typically characterized by queries and user sessions. If a CMS is unable to handle this load, then administrators may have to consider adding more Central Management Servers, so that the workload can be uniformly distributed across all the CMSs. Before taking such strategic decisions however, an administrator should have a clear idea of the current workload of the CMS, should forecast future load trends, and should accurately judge whether/not the CMS can handle its current and future load.

Moreover, user load is also a key indicator of license consumption on the CMS. By measuring the number and nature of users logging into the CMS, administrators can track license usage and plan future license requirements.

Hence, to measure the workload of the CMS from time to time and to understand its many implications, administrators should take the help of the **Central Management Server Workload** test helps.

This test reports the number and type of sessions that are currently active on the CMS. This way, the test not only highlights the current session load on CMS, but also throws light on license usage. In addition, it reveals the rate at which queries are executed and users login to the CMS, thus enabling administrators to gauge how load will change in time to come. Moreover, the test also measures how quickly the CMS responds to queries and commit operations, thereby indicating how well the CMS is handling its workload. With the help of these metrics, administrators can do the following:

- Decide whether/not more licenses need to be purchased;
- Figure out whether additional CMSs are necessary for improved performance or whether it would suffice to tune the existing CMS and its database.

Purpose	Reports the number and type of sessions that are currently active on the CMS. This way, the test not only highlights the current session load on CMS, but also throws light on license usage. In addition, it reveals the rate at which queries are executed and users login to the CMS, thus enabling administrators to gauge how load will change in time to come. Moreover, the test also measures how quickly the CMS responds to queries and commit operations, thereby indicating how well the CMS is handling its workload. With the help of these metrics, administrators can do the following: • Decide whether/not more licenses need to be purchased; • Figure out whether additional CMSs are necessary for improved performance or whether it would suffice to tune the existing CMS and its database.
Target of the test	A SAP BOBI node
Agent deploying the test	An internal/remote agent

Configurable parameters for the test	1. **TEST PERIOD** - How often should the test be executed 2. **HOST** - Host name of the server for which the test is to be configured 3. **PORT** - Enter the port to which the specified **HOST** listens. This should be the port at which the web application server hosting SAP BOBI listens. 4. **JMX REMOTE PORT** - Specify the RMI port number of the BOBI monitoring application. To know the RMI port number of the monitoring application, refer to Section 1.1.1 of this document. 5. **JNDI NAME** - Specify the lookup name for connecting to the JMX connector of the BOBI monitoring application. To know the JNDI name, refer to Section 1.1.1 of this document. 6. **JMX USER** and **JMX PASSWORD** – Enter the credentials of an enterprise authenticated BOBI user belonging to the default *monitoring users* group. 7. **CONFIRM PASSWORD** - Confirm the password by retyping it here. 8. **NODE NAME** – Specify the name of the BOBI node being monitored.		
Outputs of the test	One set of results for the CMS running in the node monitored		
Measurements made by the test	**Measurement**	**Measurement Unit**	**Interpretation**
	Number of sessions established by concurrent users: Indicates the number of sessions established by users with concurrent user licensing.	Number	By setting appropriate thresholds as per purchased Concurrent Session Based Licenses(CSBL), administrators can detect if a large number of users have logged in and if there is a possibility for subsequent users to be denied system access. In this case, administrators can plan for procuring additional concurrent user licenses.
	Number of sessions established by named users: Indicates the number of sessions established by users with named user licensing.	Number	By setting appropriate thresholds as per purchased named licenses, administrators can detect if a large number of users have logged in and if there is a possibility for subsequent users to be denied system access. In this case, administrators can plan for procuring additional named user licenses.
	Number of sessions established by servers: Indicates the number of concurrent sessions that BI platform servers have created with the CMS.	Number	If more than 250 sessions have been created by these servers on an average, then an additional CMS may have to created.
	Number of sessions established by all users: Indicates the total number of sessions established by users.	Number	This is a good indicator of the user load on the CMS.

	Average commit response time since startup: Indicates the average time taken by the CMS to perform commit operations in the CMS system database since the server was started.	Milliseconds	The general rule of thumb is that further analysis and optimization is required if the commit response time starts to exceed 750 milliseconds.
	Average query response time since startup: Indicates the average time taken by the CMS to perform query operations in the CMS system database since the server was started.	Milliseconds	The general rule of thumb is that further analysis and optimization is required if the query response time starts to exceed 120000 milliseconds.
	Commit rate: Indicates the rate of commits to the CMS system database in the last measure period.	Commits/Sec	
	Query rate: Indicates the rate of queries executed in the CMS system database in the last measure period.	Queries/Sec	Query rate directly corresponds to the load on the CMS
	Logon rate: Indicates the rate of user logons to the CMS in the last measure period.	Logins/Sec	User logon rate directly corresponds to the load on the CMS.

1.4.6 Event Server Test

The Event Server monitors the system for events, which can act as a trigger for running a report. When you set up an event trigger, the Event Server monitors the condition and notifies the CMS that an event has occurred. The CMS can then start any jobs that are set to run upon the event. The Event Server manages file-based events that occur in the storage tier.

If the Event Server is unavailable or slow, the CMS will receive no intimation regarding the occurrence of an event or may receive intimations rather late. As a result, critical jobs that CMS is configured to start upon the occurrence of an event may either not start at all or may take too long to begin. This can have serious repercussions on the performance of the SAP BOBI platform. To avoid this, administrators will have to run the **Event Server** test at pre-configured intervals. This test monitors the availability and thread pool size of the Event Server continuously, so that the failure of the server and bottlenecks in event processing (owing to improper thread pool sizing) can be captured quickly and addressed.

Purpose	Monitors the availability and thread pool size of the Event Server continuously, so that the failure of the server and bottlenecks in event processing (owing to improper thread pool sizing) can be captured quickly and addressed.

Target of the test	A SAP BOBI node		
Agent deploying the test	An internal/remote agent		
Configurable parameters for the test	1. **TEST PERIOD** - How often should the test be executed 2. **HOST** - Host name of the server for which the test is to be configured 3. **PORT** - Enter the port to which the specified **HOST** listens. This should be the port at which the web application server hosting SAP BOBI listens. 4. **JMX REMOTE PORT** - Specify the RMI port number of the BOBI monitoring application. To know the RMI port number of the monitoring application, refer to Section 1.1.1 of this document. 5. **JNDI NAME** - Specify the lookup name for connecting to the JMX connector of the BOBI monitoring application. To know the JNDI name, refer to Section 1.1.1 of this document. 6. **JMX USER** and **JMX PASSWORD** – Enter the credentials of an enterprise authenticated BOBI user belonging to the default *monitoring users* group. 7. **CONFIRM PASSWORD** - Confirm the password by retyping it here. 8. **NODE NAME** – Specify the name of the BOBI node being monitored. 9. **DETAILED DIAGNOSIS** - To make diagnosis more efficient and accurate, the eG Enterprise suite embeds an optional detailed diagnostic capability. With this capability, the eG agents can be configured to run detailed, more elaborate tests as and when specific problems are detected. To enable the detailed diagnosis capability of this test for a particular server, choose the On option. To disable the capability, click on the Off option. The option to selectively enable/disable the detailed diagnosis capability will be available only if the following conditions are fulfilled: • The eG manager license should allow the detailed diagnosis capability • Both the normal and abnormal frequencies configured for the detailed diagnosis measures should not be 0.		
Outputs of the test	One set of results for the Event server running in the node monitored		
Measurements made by the	**Measurement**	**Measurement Unit**	**Interpretation**

<table>
<tr>
<td>test</td>
<td>Health state:
Indicates the current health state of the Event server in the monitored node.</td>
<td></td>
<td>The values that this measure can report and their corresponding numeric values are discussed in the table below:
<table>
<tr><th>Measure Value</th><th>Numeric Value</th></tr>
<tr><td>Danger</td><td>0</td></tr>
<tr><td>Caution</td><td>1</td></tr>
<tr><td>Healthy</td><td>2</td></tr>
</table>
If the server is not healthy, then the detailed diagnosis of this measure, if enabled, provides the process ID of the server.
Note:
By default, this measure reports the Measure Values listed in the table above to indicate the health state of the server. In the graph of this measure however, the same is represented using the numeric values only.</td>
</tr>
<tr>
<td></td>
<td>Server running state:
Indicates the current running state of the Event server.</td>
<td></td>
<td>The values that this measure can report and their corresponding numeric values are discussed in the table below:
<table>
<tr><th>Measure Value</th><th>Numeric Value</th></tr>
<tr><td>STOPPED</td><td>0</td></tr>
<tr><td>STARTING</td><td>1</td></tr>
<tr><td>INITIALIZING</td><td>2</td></tr>
<tr><td>RUNNING</td><td>3</td></tr>
<tr><td>STOPPING</td><td>4</td></tr>
<tr><td>FAILED</td><td>5</td></tr>
<tr><td>RUNNING_WITH_ERRORS</td><td>6</td></tr>
<tr><td>RUNNING_WITH_WARNINGS</td><td>7</td></tr>
</table>
Note:
By default, this measure reports the Measure Values listed in the table above to indicate the running state of the server. In the graph of this measure however, the same is represented using the numeric values only.</td>
</tr>
</table>

<table>
<tr>
<td></td>
<td>Server enabled state:

Indicates whether/not the Event server is enabled.</td>
<td></td>
<td>The values that this measure can report and their corresponding numeric values are discussed in the table below:

<table><tr><th>Measure Value</th><th>Numeric Value</th></tr><tr><td>Disabled</td><td>0</td></tr><tr><td>Enabled</td><td>1</td></tr></table>
Note:

By default, this measure reports the Measure Values listed in the table above to indicate whether/not the server is enabled. In the graph of this measure however, the same is represented using the numeric values only.</td>
</tr>
<tr>
<td></td>
<td>Busy server threads:

Indicates the number of server threads that are currently servicing requests.</td>
<td>Number</td>
<td>If this measure reaches the configured maximum thread pool size for the server, new requests to the server would have to wait until a server thread becomes free. If this happens often, it may significantly slowdown request processing by the server. In such a situation, you may want to consider resizing the thread pool.</td>
</tr>
<tr>
<td></td>
<td>Current number of auditing events queued:

Indicates the number of auditing events that the Event server has recorded, but which have not yet been retrieved by the CMS Auditor.</td>
<td>Number</td>
<td>If this number increases without bound, it could indicate that auditing has not been configured properly or that the system is heavily loaded and generating auditing events faster than the auditor can retrieve them. When stopping servers, it is advisable to disable them first and wait for auditing events to be fully retrieved and this queue becomes empty. Otherwise, they may be retrieved only when this server has been restarted and the CMS polls for them.</td>
</tr>
</table>

1.5 The BOBI Web Access Layer

The **HTTP** test mapped to this layer emulates an HTTP access to the BI launch pad, and reports the availability and responsiveness of the launch pad to HTTP requests. BI launch pad (formerly InfoView) is a web-based interface that end users access to view, schedule, and keep track of published business intelligence (BI) reports. Since the **HTTP** test has already been discussed in the *Monitoring Web Servers* document, this document does not talk about that test again.

In addition to the **HTTP** test, the layer also monitors the health and performance of the Web Application Container Server and reports anomalies (if any). Log files generated by those server types that are responsible for data transfer and communication are also monitored by this layer, and errors in server performance are instantly captured.

Figure 1. 11: The tests mapped to the BOBI Web Access layer

1.5.1 Web Application Container Server Test

Web Application Container Servers (WACS) provide a platform for hosting SAP BusinessObjects Business Intelligence platform web applications. For example, a Central Management Console (CMC) can be hosted on a WACS. If this container crashes, then all web applications it hosts will be rendered inaccessible to users, thus adversely impacting end-user operations. In the same way, if this container is not sized with enough processing power to handle the requests it receives, request processing will significantly slowdown, resulting in scores of unhappy users once again. To steer clear off such negativities, administrators should proactively detect the inaccessibility/slowness of the WACS, accurately isolate the root-cause of the same, and promptly fix it. This is exactly where the **Web Application Container Server** test helps. This test frequently runs status checks on the WACS and brings any sudden change in state to the immediate attention of the administrator. Likewise, the test also tracks the thread pool usage by the WACS and reveals whether/not the pool contains sufficient threads for the current and future use of the WACS. If not, administrators are alerted to the inadequate pool size, so that they can rapidly increase the pool size to avert potential processing bottlenecks.

Purpose	Frequently runs status checks on the WACS and brings any sudden change in state to the immediate attention of the administrator. Likewise, the test also tracks the thread pool usage by the WACS and reveals whether/not the pool contains sufficient threads for the current and future use of the WACS. If not, administrators are alerted to the inadequate pool size, so that they can rapidly increase the pool size to avert potential processing bottlenecks.
Target of the test	A SAP BOBI node
Agent deploying the test	An internal/remote agent

Configurable parameters for the test	1. **TEST PERIOD** - How often should the test be executed 2. **HOST** - Host name of the server for which the test is to be configured 3. **PORT** - Enter the port to which the specified **HOST** listens. This should be the port at which the web application server hosting SAP BOBI listens. 4. **JMX REMOTE PORT** - Specify the RMI port number of the BOBI monitoring application. To know the RMI port number of the monitoring application, refer to Section 1.1.1 of this document. 5. **JNDI NAME** - Specify the lookup name for connecting to the JMX connector of the BOBI monitoring application. To know the JNDI name, refer to Section 1.1.1 of this document. 6. **JMX USER** and **JMX PASSWORD** – Enter the credentials of an enterprise authenticated BOBI user belonging to the default *monitoring users* group. 7. **CONFIRM PASSWORD** - Confirm the password by retyping it here. 8. **NODE NAME** – Specify the name of the BOBI node being monitored. 9. **DETAILED DIAGNOSIS** - To make diagnosis more efficient and accurate, the eG Enterprise suite embeds an optional detailed diagnostic capability. With this capability, the eG agents can be configured to run detailed, more elaborate tests as and when specific problems are detected. To enable the detailed diagnosis capability of this test for a particular server, choose the On option. To disable the capability, click on the Off option. The option to selectively enable/disable the detailed diagnosis capability will be available only if the following conditions are fulfilled: • The eG manager license should allow the detailed diagnosis capability • Both the normal and abnormal frequencies configured for the detailed diagnosis measures should not be 0.		
Outputs of the test	One set of results for the WACS running in the node monitored		
Measurements made by the	**Measurement**	**Measurement Unit**	**Interpretation**

<table>
<tr>
<td>test</td>
<td>Health state:

Indicates the current health state of the WACS in the monitored node.</td>
<td></td>
<td>The values that this measure can report and their corresponding numeric values are discussed in the table below:

<table><tr><th>Measure Value</th><th>Numeric Value</th></tr><tr><td>Danger</td><td>0</td></tr><tr><td>Caution</td><td>1</td></tr><tr><td>Healthy</td><td>2</td></tr></table>
If the server is not healthy, then the detailed diagnosis of this measure, if enabled, provides the process ID of the server.

Note:

By default, this measure reports the Measure Values listed in the table above to indicate the health state of the server. In the graph of this measure however, the same is represented using the numeric values only.</td>
</tr>
<tr>
<td></td>
<td>Server running state:

Indicates the current running state of the WACS.</td>
<td></td>
<td>The values that this measure can report and their corresponding numeric values are discussed in the table below:

<table><tr><th>Measure Value</th><th>Numeric Value</th></tr><tr><td>STOPPED</td><td>0</td></tr><tr><td>STARTING</td><td>1</td></tr><tr><td>INITIALIZING</td><td>2</td></tr><tr><td>RUNNING</td><td>3</td></tr><tr><td>STOPPING</td><td>4</td></tr><tr><td>FAILED</td><td>5</td></tr><tr><td>RUNNING_WITH_ERRORS</td><td>6</td></tr><tr><td>RUNNING_WITH_WARNINGS</td><td>7</td></tr></table>
Note:

By default, this measure reports the Measure Values listed in the table above to indicate the running state of the server. In the graph of this measure however, the same is represented using the numeric values only.</td>
</tr>
</table>

	Server enabled state: Indicates whether/not the WACS is enabled.		The values that this measure can report and their corresponding numeric values are discussed in the table below: **Measure Value** / **Numeric Value** Disabled / 0 Enabled / 1 **Note:** By default, this measure reports the **Measure Value**s listed in the table above to indicate whether/not the server is enabled. In the graph of this measure however, the same is represented using the numeric values only.
	Busy server threads: Indicates the number of server threads that are currently servicing requests.	Number	If this measure reaches the configured maximum thread pool size for the server, new requests to the server would have to wait until a server thread becomes free. If this happens often, it may significantly slowdown request processing by the server. In such a situation, you may want to consider resizing the thread pool.

1.5.2 Communication Logs Test

This test monitors logs from server types that deal with data transfer and communication in general. These server types include : Connection servers, File repository servers, Web Application container servers, Web Intelligence servers and Server Intelligence agents. The test scans these logs for specific patterns of messages and reports the count of eror and general information messages that match the configured patterns. This way, the test pinpoints critical errors that the BO BI platform that may have experienced recently and reveals the services that were affected.

Purpose	Scans the Connection servers, File repository servers, Web Application container servers, Web Intelligence servers and Server Intelligence agents logs for specific patterns of messages and reports the count of eror and general information messages that match the configured patterns. This way, the test pinpoints critical errors that the BO BI platform that may have experienced recently and reveals the services that were affected.
Target of the test	A SAP BOBI node
Agent deploying the test	An internal/remote agent

Configurable parameters for the test	1. **TEST PERIOD** - How often should the test be executed 2. **HOST** - Host name of the server for which the test is to be configured 3. **PORTNO** - Enter the port to which the specified **HOST** listens 4. **LOG DIRECTORY** - This is the directory to which logs from various nodes installed on a host are written. Typically, these logs are written to '*.glf' files in the 'logging' directory of the BOBI installation. If SAP BOBI is installed in the C drive of a Windows host, the *logging* directory will be available in the following location by default: *C:|Program Files (x86)|SAP BusinessObjects|SAP BusinessObjects Enterprise XI 4.0|* 5. **SERVER ABBREVIATIONS** - Log file names are generally of the following fomat *: <server abbreviation>_<node name>.<server type>*.glf*. For e.g., *cms_SAPBOBI.CentralManagementServer_trace.001284* is one of the log trace files from the Central Management Server running in the node called SAPBOBI. Server abbreviation in this case is **cms**. The default value for this parameter has hence been set as a comma separated list of server descriptions and their abbreviations as follows : *<server description>:<server abbreviation>*. For trace files of the Connections servers, File repository servers, Web Application container servers, Web Intelligence servers and Server Intelligence agents, this parameter has been by default set as: *Connection servers:connectionserver,Service Intelligence Agent:SIA,Web Application Container Servers:wacs,Web Intelligence Servers:webiserver,File Repository Servers:fileserver*. 6. **SEARCHPATTERN** - Enter the specific patterns of messages to be monitored. The pattern should be in the following format: <PatternName>:<Pattern>, where <PatternName> is the pattern name that will be displayed in the monitor interface and <Pattern> is an expression of the form - *expr* or expr or *expr or expr*, etc. A leading '*' signifies any number of leading characters, while a trailing '*' signifies any number of trailing characters. For example, say you specify ORA:ORA-* in the **SEARCHPATTERN** text box. This indicates that "ORA" is the pattern name to be displayed in the monitor interface. "ORA-*" indicates that the test will monitor only those lines in the log file which start with the term "ORA-". Similarly, if your pattern specification reads: offline:*offline, then it means that the pattern name is offline and that the test will monitor those lines in the log file which end with the term offline. A single pattern may also be of the form e1+e2, where + signifies an OR condition. That is, the <PatternName> is matched if either e1 is true or e2 is true. Multiple search patterns can be specified as a comma-separated list. For example: ORA:ORA-*,offline:*offline*,online:*online. Each of these patterns will be searched for in every log file that is present in the configured **LOG DIRECTORY**. 7. **LINES** - Specify two numbers in the format x:y. This means that when a line in the log file matches a particular pattern, then x lines before the matched line and y lines after the matched line will be reported in the detail diagnosis output (in addition to the matched line). The default value here is 0:0. Multiple entries can be provided as a comma-separated list. If you give 1:1 as the value for **LINES**, then this value will be applied to all the patterns specified in the **SEARCHPATTERN** field. If you give 0:0,1:1,2:1 as the value for **LINES** and if the corresponding value in the **SEARCHPATTERN** field is like ORA:ORA-*,offline:*offline*,online:*online then: 0:0 will be applied to ORA:ORA-* pattern 1:1 will be applied to offline:*offline* pattern 2:1 will be applied to online:*online pattern

8. **EXCLUDEPATTERN** - Provide a comma-separated list of patterns to be excluded from monitoring in the **EXCLUDEPATTERN** text box. For example **critical*,*exception**. By default, this parameter is set to 'none'.

9. **UNIQUEMATCH** - By default, the **UNIQUEMATCH** parameter is set to FALSE, indicating that, by default, the test checks every line in the log file for the existence of each of the configured **SEARCHPATTERNS**. By setting this parameter to **TRUE**, you can instruct the test to ignore a line and move to the next as soon as a match for one of the configured patterns is found in that line. For example, assume that Pattern1:*fatal*,Pattern2:*error* is the **SEARCHPATTERN** that has been configured. If **UNIQUEMATCH** is set to **FALSE**, then the test will read every line in the log file completely to check for the existence of messages embedding the strings 'fatal' and 'error'. If both the patterns are detected in the same line, then the number of matches will be incremented by 2. On the other hand, if **UNIQUEMATCH** is set to **TRUE**, then the test will read a line only until a match for one of the configured patterns is found and not both. This means that even if the strings 'fatal' and 'error' follow one another in the same line, the test will consider only the first match and not the next. The match count in this case will therefore be incremented by only 1.

10. **ROTATINGFILE** - This flag governs the display of descriptors for this test in the eG monitoring console.

    If this flag is set to true, the descriptors of this test will be displayed in the following format: *<Full_Path_to_LogDirectory>:<ServerAbbreviation>*. For instance, if the **LOG DIRECTORY** parameter is set to *c:|SAPBOBI|logs* and **ROTATINGFILE** is set to **true**, then, your descriptor will be of the following format: *c:|SAPBOBI|logs:<ServerAbbreviation>*. On the other hand, if the **ROTATINGFILE** flag had been set to **false**, then the descriptors will be of the following format: *<LogDirectory>:<ServerAbbreviation>* - i.e., *logs:<ServerAbbreviation>* in the case of the example above.

11. **USEUTF8** - If UTF-8 encoding is to be used for reading the log files in the configured **LOG DIRECTORY**, then, set the USEUTF8 flag to true. By default, this flag is set to false.

12. **USEUTF16** - If UTF-16 encoding is to be used for reading the log files in the configured **LOG DIRECTORY**, then, set the USEUTF16 flag to true. By default, this flag is set to false.

13. **ENCODEFORMAT** – By default, this is set to none, indicating that no encoding format applies by default. However, if the test has to use a specific encoding format for reading from the log files in the **LOG DIRECTORY**, then you will have to provide a valid encoding format here - eg., UTF-8.

14. **NODE NAME** – Specify the name of the BOBI node being monitored.

	15. **DD FREQUENCY** - Refers to the frequency with which detailed diagnosis measures are to be generated for this test. The default is 1:1. This indicates that, by default, detailed measures will be generated every time this test runs, and also every time the test detects a problem. You can modify this frequency, if you so desire. Also, if you intend to disable the detailed diagnosis capability for this test, you can do so by specifying none against **DD FREQUENCY**. 16. **DETAILED DIAGNOSIS** - To make diagnosis more efficient and accurate, the eG Enterprise suite embeds an optional detailed diagnostic capability. With this capability, the eG agents can be configured to run detailed, more elaborate tests as and when specific problems are detected. To enable the detailed diagnosis capability of this test for a particular server, choose the On option. To disable the capability, click on the Off option. The option to selectively enable/disable the detailed diagnosis capability will be available only if the following conditions are fulfilled: • The eG manager license should allow the detailed diagnosis capability • Both the normal and abnormal frequencies configured for the detailed diagnosis measures should not be 0.		
Outputs of the test	One set of results for the **SERVER ABBREVIATION** configured		
Measurements made by the test	**Measurement**	**Measurement Unit**	**Interpretation**
	High importance messages: Indicates the number of log messages with importance set to 'high' in the last measure period.	Messages	High importance designation for the log message is characterized by the '>=' symbol. For each such message, detailed diagnosis provides the details such as location (message source), timestamp, trace category, server, message, user etc.
	Highest importance messages: Indicates the number of log messages with importance set as 'highest' in the last measure period.	Messages	Highest importance designation for the log message is characterized by the '>>' symbol. For each such message, detailed diagnosis provides the details such as location (message source), timestamp, trace category, server, message, user etc.,
	Errors: Indicates the number of error log messages in the last measure period.	Messages	Error log messages have Severity values set to 'E'. For each such message, detailed diagnosis provides the details such as Location (message source), timestamp, trace category, server, message, user etc.
	Asserts: Indicates the number of Assert log messages in the last measure period.	Messages	Assert log messages have Severity values set to 'A'. For each such message, detailed diagnosis provides the details such as location (message source), timestamp, trace

			category, server, message, user etc.,
	Fatal messages: Indicates the number of fatal log messages in the last measure period.	Messages	Fatal log messages have severity values set to 'F'. For each such message, detailed diagnosis provides the details such as location (message source), timestamp, trace category, server, message, user etc.,
	New messages: Indicates the number of log messages in the last measure period.	Messages	
	Problem messages rate: Indicates the rate of all problem messages in the last measure period.	Messages/Sec	A high value is a cause for concern as it indicates that problems are occurring frequently. Compare the value of this measure across **SERVER ABBREVIATIONS** to identify the server that is the most problem-prone.

1.6 The BOBI Services Layer

This layer tracks the status and performance of all the key services associated with the servers running in the monitored node. The health of the node as a whole is revealed and the results returned by the tests executed by the Monitoring Application's probes are also reported.

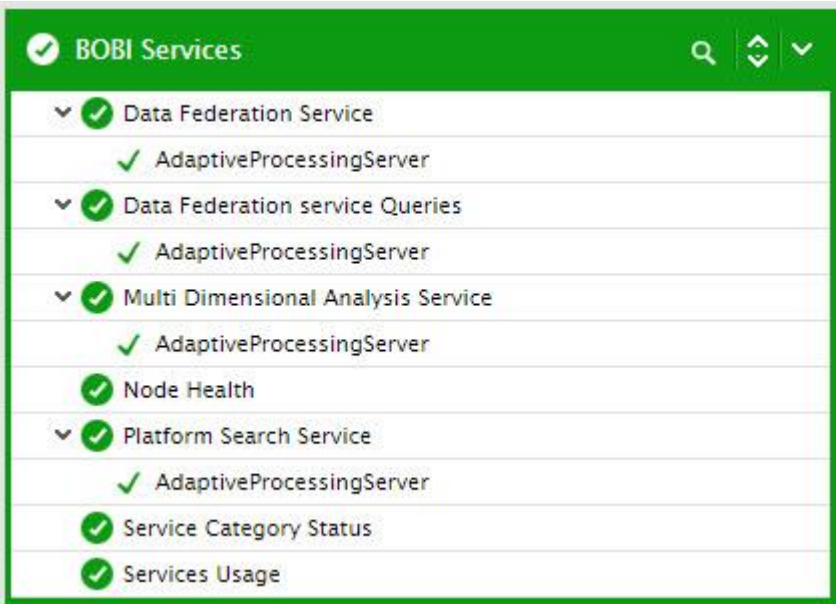

Figure 1. 12: The tests mapped to the BOBI Services Layer

1.6.1 Multi Dimensional Analysis Service Test

The **Multi-Dimensional Analysis Service (MDAS)** runs on the Adaptive Processing Server, provides access to multi-dimensional Online Analytical Processing (OLAP) data, and converts the raw data into XML, so it can be rendered into Excel, PDF, or SAP BusinessObjects Analysis (formerly Voyager) crosstabs and charts. By monitoring the session load on this service, administrators can assess the load-handling ability of the service and also determine whether/not the server and service attributes need to be fine-tuned to maximize performance. This is exactly what the **Multi Dimensional Analysis Service** test helps administrators achieve. This test tracks the MDAS sessions to the service and reports the number of OLAP data requests received through those sessions that are yet to be processed by the service. In the process, the test reveals how well the service is handling the requests it receives and quickly leads administrators to processing bottlenecks that the service may be experiencing.

Purpose	Tracks the MDAS sessions to the service and reports the number of OLAP data requests received through those sessions that are yet to be processed by the service. In the process, the test reveals how well the service is handling the requests it receives and quickly leads administrators to processing bottlenecks that the service may be experiencing.
Target of the test	A SAP BOBI node
Agent deploying the test	An internal/remote agent

<table>
<tr><td>Configurable parameters for the test</td><td colspan="3">1. TEST PERIOD - How often should the test be executed
2. HOST - Host name of the server for which the test is to be configured
3. PORT - Enter the port to which the specified HOST listens. This should be the port at which the web application server hosting SAP BOBI listens.
4. JMX REMOTE PORT - Specify the RMI port number of the BOBI monitoring application. To know the RMI port number of the monitoring application, refer to Section 1.1.1 of this document.
5. JNDI NAME - Specify the lookup name for connecting to the JMX connector of the BOBI monitoring application. To know the JNDI name, refer to Section 1.1.1 of this document.
6. JMX USER and JMX PASSWORD – Enter the credentials of an enterprise authenticated BOBI user belonging to the default monitoring users group.
7. CONFIRM PASSWORD - Confirm the password by retyping it here.
8. NODE NAME – Specify the name of the BOBI node being monitored.</td></tr>
<tr><td>Outputs of the test</td><td colspan="3">One set of results for the Multi-Dimensional Analysis Service running in the node monitored</td></tr>
<tr><td rowspan="3">Measurements made by the test</td><td>Measurement</td><td>Measurement Unit</td><td>Interpretation</td></tr>
<tr><td>Sessions:
Indicates the number of connections from MDAS clients to the server.</td><td>Number</td><td>This is a good indicator of the current session load on the service.</td></tr>
<tr><td>Cube count:
Indicates the number of data sources that are being used to supply data to the connections that have not timed out.</td><td>Number</td><td></td></tr>
<tr><td></td><td>Query count:
Indicates the number of data requests that are open between the MDS clients and the server.</td><td>Number</td><td>Ideally, the value of this measure should be low. A consistent rise in this value could indicate that the service is not processing data requests as quickly as they are received. This could be owing to a processing bottleneck on the APS hosting the service, and hence requires further investigation.</td></tr>
</table>

1.6.2 Platform Search Service Test

Platform Search enables you to search content within the SAP BusinessObjects Business Intelligence repository. It refines the search results by grouping them into categories and ranking them in order of their relevance.

The Platform Search service is the service in the Adaptive Processing Server, which has the logic to search the BOE content. The core functions of the service are indexing and searching. Before the content becomes searchable, the content needs to be indexed.

Indexing is a continuous process that involves the following sequential tasks:

- **Crawling:** Crawling is a mechanism that polls the CMS repository and identifies objects that are published, modified, or deleted. It can be done in two ways: continuous and scheduled crawling.
- **Extracting:** Extracting is a mechanism to call the extractors based upon the document type. There is a dedicated extractor for every document type that is available in the repository. New document types can be made searchable by defining new extractor plug-ins. Each of these extractors is scalable enough to extract content from large documents that contain many records.
- **Indexing:** Indexing is a mechanism that indexes all the extracted content through a third-party library, called Apache Lucene Engine.
- **Content Store:** The content store contains information such as id, cuid, name, kind, and instance extracted from the main index in a format that can be read easily. This helps to quicken the search process.

If Platform Search fails or is slow, it could be owing to one of the following reasons:

- The Platform Search service may not be available.
- The indexing mechanism may not be running.
- Indexing may be slow.
- Extractors may have failed.

In the event of the failure or slowdown of Platform Search, administrators can use the **Platform Search Service** test to determine the exact reason for the same. This test monitors the Platform Search service, reports the availability of that service on the monitored node, and if available, reveals the status of the indexing mechanism, the rate at which documents are indexed, and whether/not any extractor has failed. This way, the test leads administrators to the root-cause of problems with Platform Search.

Purpose	Monitors the Platform Search service, reports the availability of that service on the monitored node, and if available, reveals the status of the indexing mechanism, the rate at which documents are indexed, and whether/not any extractor has failed. This way, the test leads administrators to the root-cause of problems with Platform Search.
Target of the test	A SAP BOBI node
Agent deploying the test	An internal/remote agent

<table>
<tr><td>Configurable parameters for the test</td><td colspan="3">1. TEST PERIOD - How often should the test be executed
2. HOST - Host name of the server for which the test is to be configured
3. PORT - Enter the port to which the specified HOST listens. This should be the port at which the web application server hosting SAP BOBI listens.
4. JMX REMOTE PORT - Specify the RMI port number of the BOBI monitoring application. To know the RMI port number of the monitoring application, refer to Section 1.1.1 of this document.
5. JNDI NAME - Specify the lookup name for connecting to the JMX connector of the BOBI monitoring application. To know the JNDI name, refer to Section 1.1.1 of this document.
6. JMX USER and JMX PASSWORD – Enter the credentials of an enterprise authenticated BOBI user belonging to the default monitoring users group.
7. CONFIRM PASSWORD - Confirm the password by retyping it here.
8. NODE NAME – Specify the name of the BOBI node being monitored.</td></tr>
<tr><td>Outputs of the test</td><td colspan="3">One set of results for the Platform Search service running in the node monitored</td></tr>
<tr><td>Measurements made by the test</td><td>Measurement</td><td>Measurement Unit</td><td>Interpretation</td></tr>
<tr><td></td><td>Is service available ?:
Indicates whether/not the Platform Search service is available on the monitored node.</td><td></td><td>The values that this measure can report and their corresponding numeric values are discussed in the table below:
<table><tr><th>Measure Value</th><th>Numeric Value</th></tr><tr><td>No</td><td>0</td></tr><tr><td>Yes</td><td>1</td></tr></table>
Note:
By default, this measure reports the Measure Values listed in the table above to indicate the status of the Platform Search service. In the graph of this measure however, the same is represented using the numeric values only.</td></tr>
</table>

<table>
<tr><td></td><td>Is indexing running?:
Indicates whether/not the indexing mechanism is currently operational.</td><td></td><td>The values that this measure can report and their corresponding numeric values are discussed in the table below:
<table><tr><th>Measure Value</th><th>Numeric Value</th></tr><tr><td>No</td><td>0</td></tr><tr><td>Yes</td><td>1</td></tr></table>
Note:
By default, this measure reports the Measure Values listed in the table above to indicate the status of the indexing mechanism. In the graph of this measure however, the same is represented using the numeric values only.</td></tr>
<tr><td></td><td>Successful extraction rate:
Indicates the rate at which the extractors are successfully called .</td><td>Extractions/Sec</td><td>A high value is desired for this measure.</td></tr>
<tr><td></td><td>Failed extraction rate:
Indicates the rate at which the extractors failed.</td><td>Extractions/Sec</td><td>Ideally, the value of this measure should be 0. A non-zero value indicates frequent extraction failures, which can slow down indexing and consequently, Platform Search.</td></tr>
<tr><td></td><td>Document indexed rate:
Indicates the rate at which documents are indexed.</td><td>Docs/Sec</td><td>Ideally, the value of this measure should be high. A low value indicates that indexing is slow, causing Platform Search to slowdown as well.</td></tr>
</table>

1.6.3 Data Federation Service Test

The Data Federation Service helps administrators create a unified view of an organization's data sources in a fast, flexible, and easily accessible way. It allows a single Business Objects universe to map to multiple data sources and optimally federates (i.e., integrates) queries against individual sources directly. Using the **Data Federation Service** test, you can check the availability of this service on the monitored node, track the load on the service, and measure how well the service handles the load.

Purpose	Checks the availability of this service on the monitored node, tracks the load on the service, and measures how well the service handles the load.
Target of the test	A SAP BOBI node
Agent deploying the test	An internal/remote agent

<table>
<tr><td>Configurable parameters for the test</td><td colspan="3">1. TEST PERIOD - How often should the test be executed
2. HOST - Host name of the server for which the test is to be configured
3. PORT - Enter the port to which the specified HOST listens. This should be the port at which the web application server hosting SAP BOBI listens.
4. JMX REMOTE PORT - Specify the RMI port number of the BOBI monitoring application. To know the RMI port number of the monitoring application, refer to Section 1.1.1 of this document.
5. JNDI NAME - Specify the lookup name for connecting to the JMX connector of the BOBI monitoring application. To know the JNDI name, refer to Section 1.1.1 of this document.
6. JMX USER and JMX PASSWORD – Enter the credentials of an enterprise authenticated BOBI user belonging to the default monitoring users group.
7. CONFIRM PASSWORD - Confirm the password by retyping it here.
8. NODE NAME – Specify the name of the BOBI node being monitored.</td></tr>
<tr><td>Outputs of the test</td><td colspan="3">One set of results for the Data Federation Service in the node monitored</td></tr>
<tr><td rowspan="3">Measurements made by the test</td><td>Measurement</td><td>Measurement Unit</td><td>Interpretation</td></tr>
<tr><td>Is service available ?
Indicates whether/not the data federation service is currently available.</td><td></td><td>The values that this measure can report and their corresponding numeric values are discussed in the table below:<table><tr><th>Measure Value</th><th>Numeric Value</th></tr><tr><td>No</td><td>0</td></tr><tr><td>Yes</td><td>1</td></tr></table>Note:
By default, this measure reports the Measure Values listed in the table above to indicate the status of the Data Federation service. In the graph of this measure however, the same is represented using the numeric values only.</td></tr>
<tr><td>Number of connections:
Indicates the total number of user connections to the data federation query engine.</td><td>Number</td><td>This is a good indicator of the load on the data federation service.</td></tr>
<tr><td></td><td>Number of active threads:
Indicates the number of threads used for servicing requests to the DFS.</td><td>Number</td><td>This is a good indicator of how busy the service is.</td></tr>
</table>

	Number of loaded connectors Indicates the number of connectors loaded in the service.	Number	
	Number of active connections to loaded connectors: Indicates the number of active connections to the connectors loaded in the service.	Number	

1.6.4 Dashboard Logs Test

Logs related to the Dashboards Processing and Dashboards Cache servers are periodically checked by this test, so that problem events can be quickly captured and resolved.

Purpose	Logs related to the Dashboards Processing and Dashboards Cache servers are periodically checked by this test, so that problem events can be quickly captured and resolved.
Target of the test	A SAP BOBI node
Agent deploying the test	An internal/remote agent

Configurable parameters for the test	1. **TEST PERIOD** - How often should the test be executed 2. **HOST** - Host name of the server for which the test is to be configured 3. **PORTNO** - Enter the port to which the specified **HOST** listens 4. **LOG DIRECTORY** - This is the directory to which logs from various nodes installed on a host are written. Typically, these logs are written to '*.glf' files in the 'logging' directory of the BOBI installation. If SAP BOBI is installed in the C drive of a Windows host, the *logging* directory will be available in the following location by default: *C:|Program Files (x86)|SAP BusinessObjects|SAP BusinessObjects Enterprise XI 4.0|* 5. **SERVER ABBREVIATIONS** - Log file names are generally of the following fomat *: <server abbreviation>_<node name>.<server type>*.glf*. For e.g., *cms_SAPBOBI.CentralManagementServer_trace.001284* is one of the log trace files from the Central Management Server running in the node called SAPBOBI. Server abbreviation in this case is **cms**. The default value for this parameter has hence been set as a comma separated list of server descriptions and their abbreviations as follows : *<server description>:<server abbreviation>*. For trace files of the Dashboards Processing and Cache servers, this parameter has been by default set as: *Dashboard Processing servers:xcproc,Dashboard cache servers:xccache*. 6. **SEARCHPATTERN** - Enter the specific patterns of messages to be monitored. The pattern should be in the following format: <PatternName>:<Pattern>, where <PatternName> is the pattern name that will be displayed in the monitor interface and <Pattern> is an expression of the form - *expr* or expr or *expr or expr*, etc. A leading '*' signifies any number of leading characters, while a trailing '*' signifies any number of trailing characters. For example, say you specify ORA:ORA-* in the **SEARCHPATTERN** text box. This indicates that "ORA" is the pattern name to be displayed in the monitor interface. "ORA-*" indicates that the test will monitor only those lines in the log file which start with the term "ORA-". Similarly, if your pattern specification reads: offline:*offline, then it means that the pattern name is offline and that the test will monitor those lines in the log file which end with the term offline. A single pattern may also be of the form e1+e2, where + signifies an OR condition. That is, the <PatternName> is matched if either e1 is true or e2 is true. Multiple search patterns can be specified as a comma-separated list. For example: ORA:ORA-*,offline:*offline*,online:*online. Each of these patterns will be searched for in every log file that is present in the configured **LOG DIRECTORY**. 7. **LINES** - Specify two numbers in the format x:y. This means that when a line in the log file matches a particular pattern, then x lines before the matched line and y lines after the matched line will be reported in the detail diagnosis output (in addition to the matched line). The default value here is 0:0. Multiple entries can be provided as a comma-separated list. If you give 1:1 as the value for **LINES**, then this value will be applied to all the patterns specified in the **SEARCHPATTERN** field. If you give 0:0,1:1,2:1 as the value for **LINES** and if the corresponding value in the **SEARCHPATTERN** field is like ORA:ORA-*,offline:*offline*,online:*online then: 0:0 will be applied to ORA:ORA-* pattern 1:1 will be applied to offline:*offline* pattern 2:1 will be applied to online:*online pattern

8. **EXCLUDEPATTERN** - Provide a comma-separated list of patterns to be excluded from monitoring in the **EXCLUDEPATTERN** text box. For example **critical*,*exception**. By default, this parameter is set to 'none'.

9. **UNIQUEMATCH** - By default, the **UNIQUEMATCH** parameter is set to FALSE, indicating that, by default, the test checks every line in the log file for the existence of each of the configured **SEARCHPATTERNS**. By setting this parameter to **TRUE**, you can instruct the test to ignore a line and move to the next as soon as a match for one of the configured patterns is found in that line. For example, assume that Pattern1:*fatal*,Pattern2:*error* is the **SEARCHPATTERN** that has been configured. If **UNIQUEMATCH** is set to **FALSE**, then the test will read every line in the log file completely to check for the existence of messages embedding the strings 'fatal' and 'error'. If both the patterns are detected in the same line, then the number of matches will be incremented by 2. On the other hand, if **UNIQUEMATCH** is set to **TRUE**, then the test will read a line only until a match for one of the configured patterns is found and not both. This means that even if the strings 'fatal' and 'error' follow one another in the same line, the test will consider only the first match and not the next. The match count in this case will therefore be incremented by only 1.

10. **ROTATINGFILE** - This flag governs the display of descriptors for this test in the eG monitoring console.

    If this flag is set to true, the descriptors of this test will be displayed in the following format: *<Full_Path_to_LogDirectory>:<ServerAbbreviation>*. For instance, if the **LOG DIRECTORY** parameter is set to *c:|SAPBOBI|logs* and **ROTATINGFILE** is set to **true**, then, your descriptor will be of the following format: *c:|SAPBOBI|logs:<ServerAbbreviation>*. On the other hand, if the **ROTATINGFILE** flag had been set to **false**, then the descriptors will be of the following format: *<LogDirectory>:<ServerAbbreviation>* - i.e., *logs:<ServerAbbreviation>* in the case of the example above.

11. **USEUTF8** - If UTF-8 encoding is to be used for reading the log files in the configured **LOG DIRECTORY**, then, set the USEUTF8 flag to true. By default, this flag is set to false.

12. **USEUTF16** - If UTF-16 encoding is to be used for reading the log files in the configured **LOG DIRECTORY**, then, set the USEUTF16 flag to true. By default, this flag is set to false.

13. **ENCODEFORMAT** – By default, this is set to none, indicating that no encoding format applies by default. However, if the test has to use a specific encoding format for reading from the log files in the **LOG DIRECTORY**, then you will have to provide a valid encoding format here - eg., UTF-8.

14. **NODE NAME** – Specify the name of the BOBI node being monitored.

<table>
<tr><td></td><td colspan="3">15. DD FREQUENCY - Refers to the frequency with which detailed diagnosis measures are to be generated for this test. The default is 1:1. This indicates that, by default, detailed measures will be generated every time this test runs, and also every time the test detects a problem. You can modify this frequency, if you so desire. Also, if you intend to disable the detailed diagnosis capability for this test, you can do so by specifying none against DD FREQUENCY.

16. DETAILED DIAGNOSIS - To make diagnosis more efficient and accurate, the eG Enterprise suite embeds an optional detailed diagnostic capability. With this capability, the eG agents can be configured to run detailed, more elaborate tests as and when specific problems are detected. To enable the detailed diagnosis capability of this test for a particular server, choose the On option. To disable the capability, click on the Off option.

The option to selectively enable/disable the detailed diagnosis capability will be available only if the following conditions are fulfilled:

• The eG manager license should allow the detailed diagnosis capability

• Both the normal and abnormal frequencies configured for the detailed diagnosis measures should not be 0.</td></tr>
<tr><td>Outputs of the test</td><td colspan="3">One set of results for the SERVER ABBREVIATION configured</td></tr>
<tr><td rowspan="4">Measurements made by the test</td><td>Measurement</td><td>Measurement Unit</td><td>Interpretation</td></tr>
<tr><td>High importance messages:

Indicates the number of log messages with importance set to ‘high’ in the last measure period.</td><td>Messages</td><td>High importance designation for the log message is characterized by the ‘>=’ symbol. For each such message, detailed diagnosis provides the details such as location (message source), timestamp, trace category, server, message, user etc.</td></tr>
<tr><td>Highest importance messages:

Indicates the number of log messages with importance set as ‘highest’ in the last measure period.</td><td>Messages</td><td>Highest importance designation for the log message is characterized by the ‘>>’ symbol. For each such message, detailed diagnosis provides the details such as location (message source), timestamp, trace category, server, message, user etc.,</td></tr>
<tr><td>Errors:

Indicates the number of error log messages in the last measure period.</td><td>Messages</td><td>Error log messages have Severity values set to ‘E’. For each such message, detailed diagnosis provides the details such as Location (message source), timestamp, trace category, server, message, user etc.</td></tr>
<tr><td>Asserts:

Indicates the number of Assert log messages in the last measure period.</td><td>Messages</td><td>Assert log messages have Severity values set to ‘A’. For each such message, detailed diagnosis provides the details such as location (message source), timestamp, trace category, server, message, user etc.,</td></tr>
</table>

	Fatal messages: Indicates the number of fatal log messages in the last measure period.	Messages	Fatal log messages have severity values set to 'F'. For each such message, detailed diagnosis provides the details such as location (message source), timestamp, trace category, server, message, user etc.,
	New messages: Indicates the number of log messages in the last measure period.	Messages	
	Problem messages rate: Indicates the rate of all problem messages in the last measure period.	Messages/Sec	A high value is a cause for concern as it indicates that problems are occurring frequently. Compare the value of this measure across **SERVER ABBREVIATIONS** to identify the server that is the most problem-prone.

1.6.5 Report Logs Test

Logs related to the report-centric server types – namely, Crystal Reports 2013 Processing server, Crystal Reports Processing server, Crystal Reports Cache server, and Report Application server - are periodically checked by this test, so that critical problem events can be quickly captured and resolved.

Purpose	Logs related to the report-centric server types – namely, Crystal Reports 2013 Processing server, Crystal Reports Processing server, Crystal Reports Cache server, and Report Application server - are periodically checked by this test, so that critical problem events can be quickly captured and resolved.
Target of the test	A SAP BOBI node
Agent deploying the test	An internal/remote agent

Configurable parameters for the test	1. **TEST PERIOD** - How often should the test be executed 2. **HOST** - Host name of the server for which the test is to be configured 3. **PORTNO** - Enter the port to which the specified **HOST** listens 4. **LOG DIRECTORY** - This is the directory to which logs from various nodes installed on a host are written. Typically, these logs are written to '*.glf' files in the 'logging' directory of the BOBI installation. If SAP BOBI is installed in the C drive of a Windows host, the *logging* directory will be available in the following location by default: *C:|Program Files (x86)|SAP BusinessObjects|SAP BusinessObjects Enterprise XI 4.0|* 5. **SERVER ABBREVIATIONS** - Log file names are generally of the following fomat : *<server abbreviation>_<node name>.<server type>*.glf'* For e.g., *cms_SAPBOBI.CentralManagementServer_trace.001284* is one of the log trace files from the Central Management Server running in the node called SAPBOBI. Server abbreviation in this case is **cms**. The default value for this parameter has hence been set as a comma separated list of server descriptions and their abbreviations as follows *: <server description>:<server abbreviation>*. For trace files of the Crystal Reports 2013 processing servers, Crystal Reports processing servers, Crystal Reports Cache servers and Report Application Servers, this parameter has been by default set as: *Crystal Reports 2013 Processing server:cr2013proc,Crystal Reports Processing Server:crproc,Crystal Reports Cache Server:crcache,Report Application Server:rptappserver*. 6. **SEARCHPATTERN** - Enter the specific patterns of messages to be monitored. The pattern should be in the following format: *<PatternName>:<Pattern>*, where *<PatternName>* is the pattern name that will be displayed in the monitor interface and *<Pattern>* is an expression of the form - *expr* or expr or *expr or expr*, etc. A leading '*' signifies any number of leading characters, while a trailing '*' signifies any number of trailing characters. For example, say you specify ORA:ORA-* in the **SEARCHPATTERN** text box. This indicates that "ORA" is the pattern name to be displayed in the monitor interface. "ORA-*" indicates that the test will monitor only those lines in the log file which start with the term "ORA-". Similarly, if your pattern specification reads: offline:*offline, then it means that the pattern name is offline and that the test will monitor those lines in the log file which end with the term offline. A single pattern may also be of the form e1+e2, where + signifies an OR condition. That is, the *<PatternName>* is matched if either e1 is true or e2 is true. Multiple search patterns can be specified as a comma-separated list. For example: ORA:ORA-*,offline:*offline*,online:*online. 7. **LINES** - Specify two numbers in the format x:y. This means that when a line in the log file matches a particular pattern, then x lines before the matched line and y lines after the matched line will be reported in the detail diagnosis output (in addition to the matched line). The default value here is 0:0. Multiple entries can be provided as a comma-separated list. If you give 1:1 as the value for **LINES**, then this value will be applied to all the patterns specified in the **SEARCHPATTERN** field. If you give 0:0,1:1,2:1 as the value for **LINES** and if the corresponding value in the **SEARCHPATTERN** field is like ORA:ORA-*,offline:*offline*,online:*online then: 0:0 will be applied to ORA:ORA-* pattern 1:1 will be applied to offline:*offline* pattern 2:1 will be applied to online:*online pattern

8. **EXCLUDEPATTERN** - Provide a comma-separated list of patterns to be excluded from monitoring in the **EXCLUDEPATTERN** text box. For example **critical*,*exception**. By default, this parameter is set to 'none'.

9. **UNIQUEMATCH** - By default, the **UNIQUEMATCH** parameter is set to **FALSE**, indicating that, by default, the test checks every line in the log file for the existence of each of the configured **SEARCHPATTERNS**. By setting this parameter to **TRUE**, you can instruct the test to ignore a line and move to the next as soon as a match for one of the configured patterns is found in that line. For example, assume that *Pattern1:*fatal*,Pattern2:*error** is the **SEARCHPATTERN** that has been configured. If **UNIQUEMATCH** is set to **FALSE**, then the test will read every line in the log file completely to check for the existence of messages embedding the strings 'fatal' and 'error'. If both the patterns are detected in the same line, then the number of matches will be incremented by 2. On the other hand, if **UNIQUEMATCH** is set to **TRUE**, then the test will read a line only until a match for one of the configured patterns is found and not both. This means that even if the strings 'fatal' and 'error' follow one another in the same line, the test will consider only the first match and not the next. The match count in this case will therefore be incremented by only 1.

10. **ROTATINGFILE** - This flag governs the display of descriptors for this test in the eG monitoring console.

    If this flag is set to **true**, the descriptors of this test will be displayed in the following format: *<Full_Path_to_LogDirectory>:<ServerAbbreviation>*. For instance, if the **LOG DIRECTORY** parameter is set to *c:|SAPBOBI|logs* and **ROTATINGFILE** is set to **true**, then, your descriptor will be of the following format: *c:|SAPBOBI|logs:<ServerAbbreviation>*. On the other hand, if the **ROTATINGFILE** flag had been set to **false**, then the descriptors will be of the following format: *<LogDirectory>:<ServerAbbreviation>* - i.e., *logs:<ServerAbbreviation>* in the case of the example above.

11. **USEUTF8** - If UTF-8 encoding is to be used for reading the log files in the configured **LOG DIRECTORY**, then, set the **USEUTF8** flag to **true**. By default, this flag is set to **false**.

12. **USEUTF16** - If UTF-16 encoding is to be used for reading the log files in the configured **LOG DIRECTORY**, then, set the **USEUTF16** flag to **true**. By default, this flag is set to **false**.

13. **ENCODEFORMAT** – By default, this is set to *none*, indicating that no encoding format applies by default. However, if the test has to use a specific encoding format for reading from the log files in the **LOG DIRECTORY**, then you will have to provide a valid encoding format here - eg., *UTF-8*.

	14. **ENCODEFORMAT** – By default, this is set to *none*, indicating that no encoding format applies by default. However, if the test has to use a specific encoding format for reading from the log files in the **LOG DIRECTORY**, then you will have to provide a valid encoding format here - eg., *UTF-8*. 15. **NODE NAME** – Specify the name of the BOBI node being monitored. 16. **DD FREQUENCY** - Refers to the frequency with which detailed diagnosis measures are to be generated for this test. The default is *1:1*. This indicates that, by default, detailed measures will be generated every time this test runs, and also every time the test detects a problem. You can modify this frequency, if you so desire. Also, if you intend to disable the detailed diagnosis capability for this test, you can do so by specifying *none* against **DD FREQUENCY**. 17. **DETAILED DIAGNOSIS** - To make diagnosis more efficient and accurate, the eG Enterprise suite embeds an optional detailed diagnostic capability. With this capability, the eG agents can be configured to run detailed, more elaborate tests as and when specific problems are detected. To enable the detailed diagnosis capability of this test for a particular server, choose the **On** option. To disable the capability, click on the **Off** option. The option to selectively enable/disable the detailed diagnosis capability will be available only if the following conditions are fulfilled: • The eG manager license should allow the detailed diagnosis capability • Both the normal and abnormal frequencies configured for the detailed diagnosis measures should not be 0.		
Outputs of the test	One set of results for the **SERVER ABBREVIATION** configured		
Measurements made by the test	**Measurement**	**Measurement Unit**	**Interpretation**
	High importance messages: Indicates the number of log messages with importance set to 'high' in the last measure period.	Messages	High importance designation for the log message is characterized by the '>=' symbol. For each such message, detailed diagnosis provides the details such as location (message source), timestamp, trace category, server, message, user etc.
	Highest importance messages: Indicates the number of log messages with importance set as 'highest' in the last measure period.	Messages	Highest importance designation for the log message is characterized by the '>>' symbol. For each such message, detailed diagnosis provides the details such as location (message source), timestamp, trace category, server, message, user etc.,
	Errors: Indicates the number of error log messages in the last measure period.	Messages	Error log messages have Severity values set to 'E'. For each such message, detailed diagnosis provides the details such as Location (message source), timestamp, trace category, server, message, user etc.

	Asserts: Indicates the number of Assert log messages in the last measure period.	Messages	Assert log messages have Severity values set to 'A'. For each such message, detailed diagnosis provides the details such as location (message source), timestamp, trace category, server, message, user etc.,
	Fatal messages: Indicates the number of fatal log messages in the last measure period.	Messages	Fatal log messages have severity values set to 'F'. For each such message, detailed diagnosis provides the details such as location (message source), timestamp, trace category, server, message, user etc.,
	New messages: Indicates the number of log messages in the last measure period.	Messages	
	Problem messages rate: Indicates the rate of all problem messages in the last measure period.	Messages/Sec	A high value is a cause for concern as it indicates that problems are occurring frequently. Compare the value of this measure across **SERVER ABBREVIATIONS** to identify the server that is the most problem-prone.

1.6.6 Data Federation Service Queries Test

The efficiency of the Data Federation Service can be measured by the quality of the queries the data federation administration tool helps build. By periodically checking the feedback returned by the data federation query engine, administrators can determine the number of queries that may have to be optimized before being executed. The **Data Federation Service Queries** test helps administrators perform this check at regular intervals. With the help of the metrics reported by this test, administrators can quickly figure out if there are any queries that are consuming disk space and memory resources. In addition, the test also measures the query load on the data federation query engine, reports the count of queries in various stages of processing, and thus indicates where most queries are spending time – in analysis? In optimization? In execution? This way, the test points to bottlenecks in query processing and the source of these bottlenecks.

Purpose	With the help of the metrics reported by this test, administrators can quickly figure out if there are any queries that are consuming disk space and memory resources. In addition, the test also measures the query load on the data federation query engine, reports the count of queries in various stages of processing, and thus indicates where most queries are spending time – in analysis? In optimization? In execution? This way, the test points to bottlenecks in query processing and the source of these bottlenecks.
Target of the test	A SAP BOBI node
Agent deploying the test	An internal/remote agent

Configurable parameters for the test	1. **TEST PERIOD** - How often should the test be executed 2. **HOST** - Host name of the server for which the test is to be configured 3. **PORT** - Enter the port to which the specified **HOST** listens. This should be the port at which the web application server hosting SAP BOBI listens. 4. **JMX REMOTE PORT** - Specify the RMI port number of the BOBI monitoring application. To know the RMI port number of the monitoring application, refer to Section 1.1.1 of this document. 5. **JNDI NAME** - Specify the lookup name for connecting to the JMX connector of the BOBI monitoring application. To know the JNDI name, refer to Section 1.1.1 of this document. 6. **JMX USER** and **JMX PASSWORD** – Enter the credentials of an enterprise authenticated BOBI user belonging to the default *monitoring users* group. 7. **CONFIRM PASSWORD** - Confirm the password by retyping it here. 8. **NODE NAME** – Specify the name of the BOBI node being monitored.		
Outputs of the test	One set of results for the Data Federaton query engine in the node monitored		
Measurements made by the test	**Measurement**	**Measurement Unit**	**Interpretation**
	Number of running queries: Indicates the number of running queries.	Number	The value of this measure includes both memory-consuming and non-consuming queries.
	Queries consuming memory: Indicates the number of queries consuming memory.	Number	A non-zero value for this measure indicates that one/more queries are consuming memory resources. Queries that are consuming memory excessively could be candidates for optimization. You may want to view the query plan of every query in the data federation administration tool to know how much memory each query is consuming, which operators are consuming maximum memory, and how the query can be restructured to consume less memory.
	Queries using disk: Indicates the number of queries using disk space.	Number	A non-zero value for this measure indicates that one/more queries are consuming disk space. Queries that are consuming disk space excessively could be candidates for optimization.
	Queries waiting for resources: Indicates the number of running queries currently waiting for execution.	Number	

	Failed queries: Indicates the number of queries that failed.	Number	Ideally, the value of this measure should be 0. A non-zero value could indicate that one/more queries have failed. You can use the **Query Monitoring** tab of the data federation administration tool to view the exception that caused the query to fail and to troubleshoot the failure.
	Queries in query analyze step: Indicates the number of running queries currently in analyze step.	Number	If, at any given point in time, the number of queries being analyzed are way more than the count of queries being executed, it could mean that queries are spending too much time in the analysis step.
	Queries in query optimization step: Indicates the number of running queries currently in optimization step.	Number	If, at any given point in time, the number of queries being optimized are way more than the count of queries being executed, it could mean that too many inefficient/unoptimized queries are running.
	Queries in query exec step: Indicates the number of running queries currently in the execution step.	Number	
	Data output by query execution: Indicates the amount of data output by the executed queries.	MB	
	Data transferred from data sources: Indicates the amount of data read from data sources.	MB	
	Memory used by query execution: Indicates the total amount of memory used by running queries.	MB	A high value is indicative of high memory consumption by one/more queries. Queries that are consuming memory excessively could be candidates for optimization. You may want to view the query plan of every query in the data federation administration tool to know how much memory each query is consuming, which operators are consuming maximum memory, and how the query can be restructured to consume less memory.

	Disk used by query execution: Indicates the total amount of disk space used by running queries.	MB	A high value is indicative of high disk space usage by one/more queries. Queries that are consuming disk space excessively could be candidates for optimization.
	Memory used by cache: Indicates the amount of memory used for caching metadata, statistics and connectors configuration.	MB	

1.6.7 Node Health Test

A node is a group of SAP BusinessObjects Business Intelligence platform servers that run on the same host and are managed by the same Server Intelligence Agent (SIA). The first sign of problems with one/more servers running on a node is when the node is in the 'Danger' or in the 'Caution' state. Administrators should hence track node health continuously to spot server-related problems quickly. The **Node Health** test helps administrators achieve this by checking on the health of the monitored node frequently and reporting abnormalities instantly.

Purpose	Checks on the health of the monitored node frequently and reports abnormalities instantly		
Target of the test	A SAP BOBI node		
Agent deploying the test	An internal/remote agent		
Configurable parameters for the test	1. **TEST PERIOD** - How often should the test be executed 2. **HOST** - Host name of the server for which the test is to be configured 3. **PORT** - Enter the port to which the specified **HOST** listens. This should be the port at which the web application server hosting SAP BOBI listens. 4. **JMX REMOTE PORT** - Specify the RMI port number of the BOBI monitoring application. To know the RMI port number of the monitoring application, refer to Section 1.1.1 of this document. 5. **JNDI NAME** - Specify the lookup name for connecting to the JMX connector of the BOBI monitoring application. To know the JNDI name, refer to Section 1.1.1 of this document. 6. **JMX USER** and **JMX PASSWORD** – Enter the credentials of an enterprise authenticated BOBI user belonging to the default *monitoring users* group. 7. **CONFIRM PASSWORD** - Confirm the password by retyping it here. 8. **NODE NAME** – Specify the name of the BOBI node being monitored.		
Outputs of the test	One set of results for the node being monitored		
Measurements made by the	**Measurement**	**Measurement Unit**	**Interpretation**

<table>
<tr>
<td>test</td>
<td>Health state:

Indicates the overall health state of the monitored node.</td>
<td></td>
<td>The values that this measure can report and their corresponding numeric values are discussed in the table below:

<table>
<tr><th>Measure Value</th><th>Numeric Value</th></tr>
<tr><td>Danger</td><td>0</td></tr>
<tr><td>Caution</td><td>1</td></tr>
<tr><td>Healthy</td><td>2</td></tr>
<tr><td>Disabled</td><td>4</td></tr>
</table>

Note:

By default, this measure reports the Measure Values listed in the table above to indicate the health state of the node. In the graph of this measure however, the same is represented using the numeric values only.</td>
</tr>
</table>

.

1.6.8 Probes Test

Probes are a critical component of the SAP BOBI's Monitoring application. Probes monitor different services and simulate the different functionalities of BI platform components. By scheduling probes to run at specified intervals, the system administrator can track the availability and performance of key services provided by BI platform. If these probes run slowly, it can adversely impact the Monitoring application's ability to ascertain service availability and responsiveness in real-time. Using the **Probes** test, administrators can quickly isolate probes that are running slower than the rest. This knowledge will help administrators investigate the the cause for the slowness and eliminate it, so that the probes can deliver timely health reports to administrators.

In addition to highlighting slow probes, the **Probes** test also helps administrators track the results of the tests that probes run and in the process, pinpoints the probes where tests failed. Administrators can then use the Monitoring application to figure out which tests failed and thus identify unavailable/unresponsive services.

Purpose	Quickly pinpoints probes that are running slower than the rest and probes where tests failed
Target of the test	A SAP BOBI node
Agent deploying the test	An internal/remote agent

<table>
<tr><td>Configurable parameters for the test</td><td colspan="3">1. TEST PERIOD - How often should the test be executed
2. HOST - Host name of the server for which the test is to be configured
3. PORT - Enter the port to which the specified HOST listens. This should be the port at which the web application server hosting SAP BOBI listens.
4. JMX REMOTE PORT - Specify the RMI port number of the BOBI monitoring application. To know the RMI port number of the monitoring application, refer to Section 1.1.1 of this document.
5. JNDI NAME - Specify the lookup name for connecting to the JMX connector of the BOBI monitoring application. To know the JNDI name, refer to Section 1.1.1 of this document.
6. JMX USER and JMX PASSWORD – Enter the credentials of an enterprise authenticated BOBI user belonging to the default monitoring users group.
7. CONFIRM PASSWORD - Confirm the password by retyping it here.
8. NODE NAME – Specify the name of the BOBI node being monitored.</td></tr>
<tr><td>Outputs of the test</td><td colspan="3">One set of measures for each scheduled probe (both default and custom probes) running in the node. Default probes include ‘BI launch pad’, ‘CMS cache’, ‘CMS DB Connection’, ‘CMS logon & logoff’, ‘CMS ping’, ‘Crystal reports service (processing server) and crystal reports service (report application server)</td></tr>
<tr><td rowspan="3">Measurements made by the test</td><td>Measurement</td><td>Measurement Unit</td><td>Interpretation</td></tr>
<tr><td>Execution time:
Indicates the time taken for this probe to execute.</td><td>Secs</td><td>A high value is indicative of a slow probe. Compare the value of this measure across probes to identify the slowest probe.</td></tr>
<tr><td>Has passed ?
Indicates whether/not the tes run by this probe passed.</td><td></td><td>The values that this measure can report and their corresponding numeric values are discussed in the table below:
<table><tr><th>Measure Value</th><th>Numeric Value</th></tr><tr><td>Yes</td><td>1</td></tr><tr><td>No</td><td>0</td></tr></table>
Note:
By default, this measure reports the Measure Values listed in the table above to indicate the whether/not the test run by the node passed. In the graph of this measure however, the same is represented using the numeric values only.</td></tr>
</table>

1.6.9 Service Category Status Test

If the **Node Health** test reveals that a node is in an unhealthy state currently, administrators may want to instantly know which service running in that node is responsible for the node's poor health. The **Service Category Status** test reveals this. This test monitors the current state of each service running in a specific node and accurately points to those services that are in an abnormal state presently.

Purpose	Monitors the current state of each service running in a specific node and accurately points to those services that are in an abnormal state presently		
Target of the test	A SAP BOBI node		
Agent deploying the test	An internal/remote agent		
Configurable parameters for the test	1. **TEST PERIOD** - How often should the test be executed 2. **HOST** - Host name of the server for which the test is to be configured 3. **PORT** - Enter the port to which the specified **HOST** listens. This should be the port at which the web application server hosting SAP BOBI listens. 4. **JMX REMOTE PORT** - Specify the RMI port number of the BOBI monitoring application. To know the RMI port number of the monitoring application, refer to Section 1.1.1 of this document. 5. **JNDI NAME** - Specify the lookup name for connecting to the JMX connector of the BOBI monitoring application. To know the JNDI name, refer to Section 1.1.1 of this document. 6. **JMX USER** and **JMX PASSWORD** – Enter the credentials of an enterprise authenticated BOBI user belonging to the default *monitoring users* group. 7. **CONFIRM PASSWORD** - Confirm the password by retyping it here. 8. **NODE NAME** – Specify the name of the BOBI node being monitored.		
Outputs of the test	One set of measures for the monitored node		
Measurements made by the	**Measurement**	**Measurement Unit**	**Interpretation**

<table>
<tr>
<td>test</td>
<td>Dashboards services:

Indicates the current health state of the dashboards services.</td>
<td></td>
<td>The health of this service is governed by the Dashboards Processing and Dashboards Cache servers.

The values that this measure can report and their corresponding numeric values are discussed in the table below:

<table>
<tr><th>Measure Value</th><th>Numeric Value</th></tr>
<tr><td>Danger</td><td>0</td></tr>
<tr><td>Caution</td><td>1</td></tr>
<tr><td>Healthy</td><td>2</td></tr>
</table>

Note:

By default, this measure reports the Measure Values listed in the table above to indicate the health state of the dashboards services. In the graph of this measure however, the same is represented using the numeric values only.</td>
</tr>
<tr>
<td></td>
<td>Data federation services:

Indicates the current health state of the data federation services.</td>
<td></td>
<td>The health of this service is governed by the Adaptive Processing server.

The values that this measure can report and their corresponding numeric values are discussed in the table below:

<table>
<tr><th>Measure Value</th><th>Numeric Value</th></tr>
<tr><td>Danger</td><td>0</td></tr>
<tr><td>Caution</td><td>1</td></tr>
<tr><td>Healthy</td><td>2</td></tr>
</table>

Note:

By default, this measure reports the Measure Values listed in the table above to indicate the health state of the data federation services. In the graph of this measure however, the same is represented using the numeric values only.</td>
</tr>
</table>

<table>
<tr>
<td></td>
<td>Core services:

Indicates the current health state of the core services.</td>
<td></td>
<td>The health of this service is governed by the Central Management Server, Adaptive Processing Server, Event Server, Adaptive Job Server, Input File Repository Server, Web Application Container Server and Output File Repository Server.

The values that this measure can report and their corresponding numeric values are discussed in the table below:

<table>
<tr><th>Measure Value</th><th>Numeric Value</th></tr>
<tr><td>Danger</td><td>0</td></tr>
<tr><td>Caution</td><td>1</td></tr>
<tr><td>Healthy</td><td>2</td></tr>
</table>

Note:

By default, this measure reports the Measure Values listed in the table above to indicate the health state of the core services. In the graph of this measure however, the same is represented using the numeric values only.</td>
</tr>
<tr>
<td></td>
<td>Analysis services:

Indicates the current health state of the analysis services.</td>
<td></td>
<td>The health of this service is governed by the Adaptive Processing server.

The values that this measure can report and their corresponding numeric values are discussed in the table below:

<table>
<tr><th>Measure Value</th><th>Numeric Value</th></tr>
<tr><td>Danger</td><td>0</td></tr>
<tr><td>Caution</td><td>1</td></tr>
<tr><td>Healthy</td><td>2</td></tr>
</table>

Note:

By default, this measure reports the Measure Values listed in the table above to indicate the health state of the analysis services. In the graph of this measure however, the same is represented using the numeric values only.</td>
</tr>
</table>

<table>
<tr>
<td></td>
<td>Crystal report services:

Indicates the current health state of the Crystal report services.</td>
<td></td>
<td>The health of this service is governed by the Crystal Reports Cache Server and the Crystal Reports Processing Server 2013 version, Adaptive Job Server, Crystal Reports Report Application Server 2013 version and Crystal Reports Processing Server.

The values that this measure can report and their corresponding numeric values are discussed in the table below:

<table>
<tr><th>Measure Value</th><th>Numeric Value</th></tr>
<tr><td>Danger</td><td>0</td></tr>
<tr><td>Caution</td><td>1</td></tr>
<tr><td>Healthy</td><td>2</td></tr>
</table>

Note:

By default, this measure reports the Measure Values listed in the table above to indicate the health state of the Crystal report services. In the graph of this measure however, the same is represented using the numeric values only.</td>
</tr>
<tr>
<td></td>
<td>Lifecycle management services:

Indicates the current health state of the lifecycle management services.</td>
<td></td>
<td>The health of this service is governed by the Adaptive Processing server and the Adaptive Job server.

The values that this measure can report and their corresponding numeric values are discussed in the table below:

<table>
<tr><th>Measure Value</th><th>Numeric Value</th></tr>
<tr><td>Danger</td><td>0</td></tr>
<tr><td>Caution</td><td>1</td></tr>
<tr><td>Healthy</td><td>2</td></tr>
</table>

Note:

By default, this measure reports the Measure Values listed in the table above to indicate the health state of the lifecycle management services. In the graph of this measure however, the same is represented using the numeric values only.</td>
</tr>
</table>

<table>
<tr>
<td></td>
<td>Connectivity services:

Indicates the current health state of the connectivity services.</td>
<td></td>
<td>The health of this service is governed by the Connection Server, 32 bit Connection Server and Adaptive Processing Server.

The values that this measure can report and their corresponding numeric values are discussed in the table below:

<table>
<tr><th>Measure Value</th><th>Numeric Value</th></tr>
<tr><td>Danger</td><td>0</td></tr>
<tr><td>Caution</td><td>1</td></tr>
<tr><td>Healthy</td><td>2</td></tr>
</table>

Note:

By default, this measure reports the Measure Values listed in the table above to indicate the health state of the Connectivity services. In the graph of this measure however, the same is represented using the numeric values only.</td>
</tr>
<tr>
<td></td>
<td>Web intelligence services:

Indicates the current health state of the web intelligence services.</td>
<td></td>
<td>The health of this service is governed by the Adaptive Processing Server, Adaptive Job Server and the Web Intelligence Processing Server.

The values that this measure can report and their corresponding numeric values are discussed in the table below:

<table>
<tr><th>Measure Value</th><th>Numeric Value</th></tr>
<tr><td>Danger</td><td>0</td></tr>
<tr><td>Caution</td><td>1</td></tr>
<tr><td>Healthy</td><td>2</td></tr>
</table>

Note:

By default, this measure reports the Measure Values listed in the table above to indicate the health state of the Web intelligence services. In the graph of this measure however, the same is represented using the numeric values only.</td>
</tr>
</table>

1.6.10 Services Usage Test

This test monitors how each of the services operating in the monitored node are being used, and thus points to the most popular/busiest application of the SAP BOBI platform.

Purpose	Monitors how each of the services operating in the monitored node are being used, and thus points to the most popular/busiest application of the SAP BOBI platform		
Target of the test	A SAP BOBI node		
Agent deploying the test	An internal/remote agent		
Configurable parameters for the test	1. **TEST PERIOD** - How often should the test be executed 2. **HOST** - Host name of the server for which the test is to be configured 3. **PORT** - Enter the port to which the specified **HOST** listens. This should be the port at which the web application server hosting SAP BOBI listens. 4. **JMX REMOTE PORT** - Specify the RMI port number of the BOBI monitoring application. To know the RMI port number of the monitoring application, refer to Section 1.1.1 of this document. 5. **JNDI NAME** - Specify the lookup name for connecting to the JMX connector of the BOBI monitoring application. To know the JNDI name, refer to Section 1.1.1 of this document. 6. **JMX USER** and **JMX PASSWORD** – Enter the credentials of an enterprise authenticated BOBI user belonging to the default *monitoring users* group. 7. **CONFIRM PASSWORD** - Confirm the password by retyping it here. 8. **NODE NAME** – Specify the name of the BOBI node being monitored.		
Outputs of the test	One set of measures for the monitored node		
Measurements made by the test	**Measurement**	**Measurement Unit**	**Interpretation**
	Xcelsius models defined rate: Indicates the rate at which Xcelsius dashboard definitions were created in the last measure period.	Models/Sec	
	Publications rate: Indicates the rate of publications made in the last measurement period.	Publications/Sec	

	Crystal reports defined rate: Indicates the rate at which crystal reports were defined in the last measure period.		
	Web intelligence reports defined rate: Indicates the rate at which Web intelligence reports were defined in the last measure period.	Reports/Sec	
	Crystal reports defined rate: Indicates the rate at which crystal reports were defined in the last measure period.	Reports/Sec	
	Program instances: Indicates the rate at which program instances were executed in the last measure period.	Executions/Sec	
	Xcelcius shockwave files generated rate: Indicates the rate at which Xcelcius shockwave files were generated in the last measure period.	Files/Sec	
	Crystal reports scheduled rate: Indicates the rate at which crystal reports were scheduled during the last measure period.	Reports/Sec	

	Web intelligence reports scheduled rate : Indicates the rate at which web intelligence reports were scheduled during the last measure period.	Reports/Sec	

2

Conclusion

This document has described in detail the monitoring paradigm used and the measurement capabilities of the eG Enterprise suite of products with respect to **SAP BOBI**. For details of how to administer and use the eG Enterprise suite of products, refer to the user manuals.

We will be adding new measurement capabilities into the future versions of the eG Enterprise suite. If you can identify new capabilities that you would like us to incorporate in the eG Enterprise suite of products, please contact support@eginnovations.com. We look forward to your support and cooperation. Any feedback regarding this manual or any other aspects of the eG Enterprise suite can be forwarded to feedback@eginnovations.com.