SAP BusinessObjects

Viewing Documents Using OpenDocument

- SAP BusinessObjects Business Intelligence platform 4.1

2012-03-16

SAP

Copyright

2012-03-16

Contents

Chapter 1 Document History....5

Chapter 2 Getting started....7

2.1 About this documentation....7
2.1.1 Who should use this documentation?....7
2.1.2 About OpenDocument....7
2.2 What's new in SAP BusinessObjects Business Intelligence platform 4.0....8
2.3 Migrating your links....8
2.3.1 Changes to the default URL path....8
2.3.2 Deprecated parameters....9

Chapter 3 OpenDocument syntax....11

3.1 Basic URL syntax....11
3.2 URL syntax considerations....11

Chapter 4 Session management....13

4.1 Serialized sessions....13
4.2 Logon tokens....14
4.3 User sessions....15

Chapter 5 Parameter reference....17

5.1 Session management parameters....19
5.1.1 serSes....19
5.1.2 token....20
5.2 Document identifier parameters....21
5.2.1 iDocID....21
5.2.2 sDocName....22
5.2.3 sIDType....22
5.2.4 sInstance....23
5.3 Input parameters....24
5.3.1 lsC....24

5.3.2 lsI[NAME] 24
5.3.3 lsM[NAME] 25
5.3.4 lsR[NAME] 26
5.3.5 lsS[NAME] 27
5.3.6 sPartContext 28
5.3.7 sRefresh 29
5.3.8 sReportMode 29
5.3.9 sReportName 30
5.3.10 sReportPart 30
5.4 Output parameters 31
5.4.1 NAll 31
5.4.2 noDocument 32
5.4.3 sViewer 32
5.4.4 sOutputFormat 33

Appendix A More Information 35

Index 37

Document History

The following table provides an overview of the most important document changes.

Version	Date	Description
SAP BusinessObjects Business Intelligence platform 4.0	November, 2011	First release of this document.

Getting started

2.1 About this documentation

This documentation provides you with information for constructing parameterized URLs with the OpenDocument syntax. OpenDocument URLs link to Business Intelligence (BI) documents in an SAP BusinessObjects Business Intelligence platform system. A parameter reference, including syntax and usage examples, is provided for each OpenDocument URL parameter.

For information about deploying the OpenDocument web application after the installation of the BI platform, see the *SAP BusinessObjects Business Intelligence platform Web Application Deployment Guide*.

2.1.1 Who should use this documentation?

This documentation is for anyone creating URLs to BI documents with the OpenDocument syntax. We recommend consulting this guide if you are:

- Providing end users with hyperlinks to a document through email or other direct means.
- Embedding hyperlinks in one document to another.
- Programmatically generating hyperlinks to documents in your custom application.

Familiarity with the management and organization of objects in your BI platform deployment is beneficial.

2.1.2 About OpenDocument

OpenDocument is one of many deployed web applications within a BI platform installation. It processes incoming URL requests for documents and any other viewable object type in the Central Management Server (CMS), and delivers the correct document to the end user in the appropriate viewer. This allows you to send users direct links to a document and avoid having them navigate through a folder hierarchy, such as in BI launch pad. The OpenDocument syntax and its parameters allow you to construct URLs that link to these documents. For example, consider the following URL:

```
http://<servername>:<port>/BOE/OpenDocument/opendoc/openDocument.jsp?iDocID=Aa6GrrM79cRAmaOSMGoadKI&sIDType=CUID
```

Note:
Replace <servername>:<port> with the name and port number of your web server where OpenDocument is deployed.

This URL accesses the object in the CMS with the CUID value of `Aa6GrrM79cRAmaOSMGoadKI`. If this is a Crystal report, for example, then the report is rendered to the user in a default SAP Crystal Reports viewer. In this example, `iDocID` is one of many URL parameters. These parameters specify how to access a particular document in the CMS, or determine how to display the document to the user.

You can link to many viewable object types with the OpenDocument syntax. Some examples include:

- Crystal reports
- Web Intelligence documents
- Analysis workspaces
- BI launch pad workspaces
- Dashboards objects (formerly Xcelsius)

Some of the designers for these BI document types provide GUI-based URL builders to help you embed openDocument URLs into your documents. Consult their respective product documentation for information on these features.

2.2 What's new in SAP BusinessObjects Business Intelligence platform 4.0

serSes parameter

The `serSes` parameter can now be used to pass a serialized Enterprise session to an OpenDocument URL. Use this parameter so users avoid encountering additional logon prompts in your custom application.

For more information on the `serSes` parameter and managing user sessions, see Session management and serSes.

2.3 Migrating your links

2.3.1 Changes to the default URL path

The default URL to the OpenDocument web application bundle has changed in SAP BusinessObjects Business Intelligence platform 4.0. New absolute OpenDocument links need to use the new default URL:

```
http://<servername>:<port>/BOE/OpenDocument/opendoc/openDocument.jsp?<parameter1>&<parameter2>&...&<parameterN>
```

If you are migrating reports with existing links from an XI 3.x release platform, resolve the issue by setting up the following redirect in your web server:

- Redirect: `../OpenDocument/opendoc/openDocument.jsp`
- To: `../BOE/OpenDocument/opendoc/openDocument.jsp`

Note:

- Ensure that all URL request parameters are forwarded correctly by your redirect. Refer to your web server documentation for detailed steps on implementing a redirect.
- SAP BusinessObjects Business Intelligence platform 4.0 only supports a Java deployment of OpenDocument. The OpenDocument web bundle is part of the `BOE.war` file.

2.3.2 Deprecated parameters

This section lists deprecated and obsolete OpenDocument parameters as of SAP BusinessObjects Business Intelligence platform 4.0. Obsolete parameters are unsupported.

Note:
Deprecated and obsolete members as of SAP BusinessObjects Enterprise XI 3.1 Service Packs are also listed for reference.

Table 2-5: Deprecated Parameters

Parameter	Description	Replace with
`sIDType=GUID` **Note:** Deprecated in SAP BusinessObjects Enterprise XI 3.1 SP3)	Specifies that a GUID is used to specify the viewable document. Use in conjunction with `iDocID`.	Use `sIDType=CUID` instead.
`sIDType=RUID` **Note:** Deprecated in SAP BusinessObjects Enterprise XI 3.1 SP3)	Specifies that a RUID is used to specify the viewable document. Use in conjunction with `iDocID`.	Use `sIDType=CUID` instead.

Parameter	Description	Replace with
`sKind`	Specifies the `SI_KIND` property of the target Desktop Intelligence document.	Use `iDocID` instead.
`sPath`	The file path of the target document.	Use `iDocID` instead. **Note:** `sPath` does not support the use of localized folder names which are available in this release. Legacy documents that use `sPath` to reference the correct folder name as stored in the CMS will continue to work while under deprecation. But it is recommended that you migrate your links to use the `iDocID` parameter instead.
`sType`	Specifies the file type of the target document.	Use `iDocID` instead.
`sViewer=actx`	Specifies the Crystal Reports ActiveX Viewer.	Use `sViewer=html` or `sViewer=part` instead. The ActiveX Viewer is deprecated as of this release.
`sViewer=java`	Specifies the Crystal Reports Java Applet Viewer.	Use `sViewer=html` or `sViewer=part` instead. The Java Applet Viewer is deprecated as of this release.

Table 2-6: Obsolete Parameters

Parameter	Description	Replace with
`sWindow` **Note:** Obsolete as of SAP BusinessObjects Enterprise XI 3.1 SP3	Indicates whether the target document will open in the current browser window or whether a new window will be launched.	Use the HTML anchor's `target` attribute or an equivalent. For example: `<a href="..." target="_blank">...</a>`

OpenDocument syntax

3.1 Basic URL syntax

The basic syntax for an OpenDocument URL is as follows:

```
http://<servername>:<port>/BOE/OpenDocument/opendoc/openDocument.jsp?<parameter1>&<parameter2>&...&<parameterN>
```

Note:
Variables are denoted with angle brackets. You must substitute the proper value for these variables. For example, you must use the name of your BI platform server where OpenDocument is hosted in place of `<servername>` and you must use the correct port number in place of `<port>` to access the OpenDocument web application.

3.2 URL syntax considerations

Accessing documents

You must include the `iDocID` or `sDocName` parameter in your OpenDocument URL to specify the document to be viewed. Since there may be multiple documents in the Central Management Server (CMS) with the same name, and documents can be moved or renamed, it is recommended that you use `iDocID` to ensure uniqueness.

Joining parameters

Join parameters with the ampersand (&). Do not place spaces around the ampersand. For example:
`sType=wid&sDocName=Sales2003`

The ampersand is always required between parameters.

Spaces and special characters in parameter values

Because some browsers cannot interpret spaces, the parameters of the link cannot contain spaces or other special characters that require URL encoding. To avoid the misinterpretation of special characters, you can define a URL-encoded string in the source database to replace the special character with an escape sequence. This will allow the database to ignore the special character and correctly interpret the parameter value. Note that certain RDBMS have functions that allow you to replace one special character with another.

By creating an escape sequence for the plus sign (+), you can instruct the database to interpret the plus sign as a space. In this case, a document title Sales Report for 2003 would be specified in the DocName parameter as: `&sDocName=Sales+Report+for+2003&`

This syntax prevents the database from misinterpreting the spaces in the title.

In addition, values for serialized sessions (using the `serSes` parameter) and logon tokens (using the `token` parameter) must be URL-encoded by your application before being passed to the OpenDocument URL string.

Trailing spaces in parameter values

Trim trailing spaces at the end of parameter values and prompt names. Do not replace them with a plus sign (+). The viewer may not know whether to interpret the plus sign (+) as part of the prompt name or as a space. For example, if the prompt name displays:

```
Select a City:_
```

(where _ represents a space), enter the following text in the link:

```
lsSSelect+a+City:=Paris
```

where the spaces within the prompt name are replaced with the plus sign, and the trailing space is trimmed off.

Capitalization

All of the OpenDocument parameters and parameter values are case sensitive.

URL length limit

OpenDocument may add characters to your URL when it redirects to the requested document; however, encoded URLs cannot exceed the maximum character limit for the supported browsers. For example, certain versions of Internet Explorer limit the URL length to 2083 characters. Therefore, know the browser character limit to ensure your URL will be within the maximum limit.

Parameter values in links to sub-reports

You cannot pass parameter values to a sub-report of a target Crystal report.

Opening a new window

To force OpenDocument HTML links to open a new browser window, use the HTML anchor's `target` attribute or an equivalent. For example:

```
<a href="http://<servername>:<port>/BOE/OpenDocument/opendoc/<platformSpecific>?iDocID=Aa6GrrM79cRAmaOSMGoadKI&sIDType=CUID" target="_blank">hyperlink text</a>
```

Session management

Normally when using an OpenDocument link to access documents secured in the BI platform, the user will be prompted for credentials. OpenDocument provides two parameters to avoid having the user prompted for their username and password information. You can either insert a serialized session or a logon token directly into the OpenDocument URL. This gives you control over the duration of the access to the document. OpenDocument URLs can be set to different languages.

4.1 Serialized sessions

Serialized sessions can be used in OpenDocument by inserting the `serSes` parameter into the OpenDocument URL. This allows users to access files without being prompted for credentials. Creating a serialized sessions does not use up an additional licence. Serialized sessions expire if the original user's session times out or logs off.

Example:

The following example uses the BI platform Java SDK to pass in a serialized session to the OpenDocument URL. For more information on the `IEnterpriseSession.getSerializedSession` method, see the *SAP BusinessObjects Business Intelligence platform Java API Reference*.

```
String openDocumentSerSes() throws SDKException, UnsupportedEncodingException
{
 IEnterpriseSession sess = CrystalEnterprise.getSessionMgr().logon ("username", "password", "<cms>:<port>",
 "secEnterprise");
 String serSession = sess.getSerializedSession();
 String serSesEncode = URLEncoder.encode(serSession, "UTF-8");
 return ("http://<server>:<port>/BOE/OpenDocument/opendoc/openDocument.jsp?iDocID=Aa6GrrM79cRAmaOSMGoad
KI&sIDType=CUID&serSes=" + serSesEncode);
}
```

Note:

- Replace <server>:<port> with the server name and port number of your web server.
- Replace <cms>:<port> with the Central Management Server (CMS) name and port number.
- You must URL-encode the serialized session.
- Since an OpenDocument URL with a serialized session contains the user session, they must not be shared for security reasons.

4.2 Logon tokens

Logon tokens can be used in OpenDocument by inserting the `token` parameter into the OpenDocument URL. Logon tokens allow users access to files secured in the BI platform without being prompted for credentials, while also giving you control on the duration of the access to the file. Creating a new logon token uses up an additional licence.

Example: Using the BI platform Java SDK

The following example uses the BI platform Java SDK to pass in a logon token to the OpenDocument URL. For more information on the `ILogonTokenMgr.createLogonToken` method, see the *SAP BusinessObjects Business Intelligence platform Java API Reference*.

```
String openDocumentToken() throws SDKException, UnsupportedEncodingException
{
 IEnterpriseSession sess = CrystalEnterprise.getSessionMgr().logon ("username", "password", "<cms>:</port>",
 "secEnterprise");
 String token = sess.getLogonTokenMgr().createLogonToken("",120,100);
 String tokenEncode = URLEncoder.encode(token, "UTF-8");
 sess.logoff();

 return ("http://<server>:<port>/BOE/OpenDocument/opendoc/openDocument.jsp?iDocID=Aa6GrrM79cRAmaOSMGoad
KI&sIDType=CUID&token=" + tokenEncode);
}
```

Note:

- Replace `<server>:<port>` with the server name and port number of your web server.
- Replace `<cms>:<port>` with the Central Management Server (CMS) name and port number.
- The `createLogonToken` method allows you to specify the machine that can use the token (which can be empty to allow any user to use the token), the number of minutes the token is valid for, and the number of logons that the token can be used for as parameters. Since the newly created logon token consumes an additional session, `sess.logoff` is called to logoff the original session.
- Since an OpenDocument URL with a logon token contains the user session, they must not be shared for security reasons.

Example: Using the BI platform RESTful Web Services SDK

The following example passes token fetched using the BI platform RESTful Web Services SDK to the OpenDocument URL. For more information see the *SAP BusinessObjects Business Intelligence platform RESTful Web Service Developer guide*.

```
http://<server>:<port>/BOE/OpenDocument/opendoc/openDocument.jsp?
iDocID=<documentID>&sIDType=CUID&token=<logonToken>
```

Note:

- A URL-encoded logon token may contain a large number of characters. Some web browsers may limit the number of characters that are allowed in a URL.
- Replace `<server>:<port>` with the server name and port number of your web server.
- Replace `<documentID>` with the ID of the document to retrieve.

- Replace `<logonToken>` with the URL-encoded logon token value retrieved using the BI platform RESTful Web Services.

4.3 User sessions

When OpenDocument is used from BI launch pad or the CMC, it will access the current user session and the user does not need to enter credentials. When a document is viewed using an OpenDocument URL, the user will be prompted for credentials except in the following cases:

- Vintela or Siteminder SSO is configured for the deployed OpenDocument web application.
- The OpenDocument URL uses a `serSes` or a `token` parameter.
- The OpenDocument application has an existing user session for that browser session.

If the existing session is different than the session in the `serSes` or `token` parameter, the existing session will be closed and a new session will be created. That is, you can use `serSes` or `token` parameter to over-ride an existing user session. The OpenDocument application will look for an existing user session in the Web application session and in cookies.

Note:
Only one OpenDocument session can be created from a single browser session.

If the new `serSes` or `token` parameter is incorrect and there is an existing user session, OpenDocument will attempt to open the document using the current user session. If it can't it will then prompt the user for credentials.

Parameter reference

This section provides details about the available OpenDocument parameters, their specific uses, and relevant examples.

Note:
The document to which an OpenDocument link points to is referred to as the target document.

Table 5-1: Session Management Parameters

Parameter	Description
serSes	Specifies a valid serialized Enterprise session.
token	Specifies a valid logon token for the current Enterprise session.

Table 5-2: Document Identifier Parameters

Parameter	Description
iDocID	Specifies the unique identifier of the viewable document in the CMS. Use in conjunction with `sIDType`.
sDocName	Specifies the name of the viewable document in the CMS.
sIDType	Specifies the type of object identifier used to specify the viewable document. Use in conjunction with `iDocID`.
sInstance	Specifies the scheduled instance of the target document to open. Use in conjunction with `sDocName` or `iDocID`.

Table 5-3: Input Parameters

Parameter	Description
lsC	Specifies a contextual prompt for Web Intelligence documents if there is an ambiguity during SQL generation.
lsM[NAME]	Specifies multiple values for a prompt. `[NAME]` is the text of the prompt.
lsR[NAME]	Specifies a range of values for a prompt. `[NAME]` is the text of the prompt.
lsS[NAME]	Specifies a value for a single prompt. `[NAME]` is the text of the prompt.
sPartContext	Specifies the data context of a Crystal report part. Use in conjunction with `sReportPart`.
sRefresh	Indicates whether a database refresh should be forced when the target document is opened.
sReportMode	Indicates whether the link should open the full target Crystal report or just the report part specified in.
sReportName	Specifies the report to open if the target document contains multiple reports.
sReportPart	Specifies the part of the target Crystal report to open.

Table 5-4: Output Parameters

Parameter	Description
NAII	Indicates whether to force the display of the prompt selection page for Interactive Analysis prompts.
sOutputFormat	Specifies the format in which to open the target document.

Parameter	Description
sViewer	Specifies the selected report viewer.
sViewer	Used with Web Intelligence reports, a value of `true` automatically forces a report to open in design mode.

5.1 Session management parameters

5.1.1 serSes

Syntax	Description	Values
`serSes`	Specifies a valid serialized Enterprise session.	A serialized string representing the current Enterprise session.

Contains a serialized session of the current user session. This can be entered into an OpenDocument URL to allow users to access files without being prompted for credentials. Creating a serialized sessions does not use up an additional licence. Serialized sessions expire if the original user's session times out or logs off.

Example:

The following example uses the BI platform Java SDK to pass in a serialized session to the OpenDocument URL. For more information on the `IEnterpriseSession.getSerializedSession` method, see the *SAP BusinessObjects Business Intelligence platform Java API Reference*. You can retrieve a serialized session in a similar fashion using other Bi platform SDKs such as .NET and Web Services.

```
String openDocumentSerSes() throws SDKException, UnsupportedEncodingException
{
 IEnterpriseSession sess = CrystalEnterprise.getSessionMgr().logon ("username", "password", "<cms>:<port>",
 "secEnterprise");
 String serSession = sess.getSerializedSession();
 String serSesEncode = URLEncoder.encode(serSession, "UTF-8");
 return ("http://<server>:<port>/BOE/OpenDocument/opendoc/openDocument.jsp?iDocID=Aa6GrrM79cRAmaOSMGoad
```

```
KI&sIDType=CUID&serSes=" + serSesEncode);
}
```

Note:

- Replace <server>:<port> with the server name and port number of your web server.
- Replace <cms>:<port> with the Central Management Server (CMS) name and port number.
- You must URL-encode the serialized session.
- Since an OpenDocument URL with a serialized session contains the user session, they must not be shared for security reasons.

5.1.2 token

Syntax	Description	Values
`token`	Specifies a valid logon token for the current Enterprise session.	The logon token for the current Enterprise session.

Contains the logon token for the current user. This can be entered into an OpenDocument URL to allow users to access files without being prompted for credentials. Creating a new logon token uses up an additional licence.

Example: Using the BI platform Java SDK

The following example uses the BI platform Java SDK to pass in a logon token to the OpenDocument URL. For more information on the `ILogonTokenMgr.createLogonToken` method, see the *SAP BusinessObjects Business Intelligence platform Java API Reference*. You can create logon tokens in a similar fashion using other BI platform SDKs such as .NET and Web Services.

```
String openDocumentToken() throws SDKException, UnsupportedEncodingException
{
 IEnterpriseSession sess = CrystalEnterprise.getSessionMgr().logon ("username", "password", "<cms>:<port>",
 "secEnterprise");
 String token = sess.getLogonTokenMgr().createLogonToken("",120,100);
 String tokenEncode = URLEncoder.encode(token, "UTF-8");
 return ("http://<server>:<port>/BOE/OpenDocument/opendoc/openDocument.jsp?iDocID=Aa6GrrM79cRAmaOSMGoad
KI&sIDType=CUID&token=" + tokenEncode);
}
```

Note:

- Replace <server>:<port> with the server name and port number of your web server.
- Replace <cms>:<port> with the Central Management Server (CMS) name and port number.
- The `createLogonToken` method allows you to specify the machine that can use the token (which can be empty to allow any user to use the token), the number of minutes the token is valid for, and the number of logons that the token can be used for as parameters.

- Since an OpenDocument URL with a logon token contains the user session, they must not be shared for security reasons.

Example: Using the BI platform RESTful Web Services SDK

The following example passes token fetched using the BI platform RESTful Web Services SDK to the OpenDocument URL. For more information see the *SAP BusinessObjects Business Intelligence platform RESTful Web Service Developer guide*.

```
http://<server>:<port>/BOE/OpenDocument/opendoc/openDocument.jsp?
iDocID=<documentID>&sIDType=CUID&token=<logonToken>
```

Note:

- A URL-encoded logon token may contain a large number of characters. Some web browsers may limit the number of characters that are allowed in a URL.
- Replace `<server>:<port>` with the server name and port number of your web server.
- Replace `<documentID>` with the ID of the document to retrieve.
- Replace `<logonToken>` with the URL-encoded logon token value retrieved using the BI platform RESTful Web Services.

5.2 Document identifier parameters

5.2.1 iDocID

Syntax	Description	Values
`iDocID`	Specifies the unique identifier of the viewable document in the CMS. Use in conjunction with `sIDType`.	A numerical identifier associated with the document in the CMS.

You must include the `iDocID` or `sDocName` parameter in your OpenDocument URL to specify the document to be viewed. Since there may be multiple documents in the CMS with the same name, it is recommended that you use `iDocID` to ensure uniqueness.

You can see identifier values for a document within the Central Management Console (CMC) or BI launch pad applications. The properties page for each document contains the document ID and the CUID. You can also obtain the identifier programmatically using the BI platform SDK. For example, in

the Java SDK the `com.crystaldecisions.sdk.occa.infostore.IInfoObject` interface contains `getID` and `getCUID` methods which you can pass to an OpenDocument URL.

Note:
If you pass in an InfoObject ID rather than a CUID, you do not need to specify the `sIDType` parameter. However, InfoObject IDs are changed when migrating documents from one CMS to another. It is recommended that the CUID be used, which is preserved during migration.

Example:

```
http://<servername>:<port>/BOE/OpenDocument/opendoc/openDocument.jsp?iDocID=2010
```

```
http://<servername>:<port>/BOE/OpenDocument/opendoc/openDocument.jsp?iDocID=Aa6GrrM79cRAmaOSMGoadKI&sID
Type=CUID
```

5.2.2 sDocName

Syntax	Description	Values
`sDocName`	Specifies the name of the viewable document in the CMS.	The title of the document in the CMS.

You must include the `iDocID` or `sDocName` parameter in your OpenDocument URL to specify the document to be viewed. Since there may be multiple documents in the CMS with the same name, and documents can be moved or renamed, it is recommended that you use `iDocID` to ensure uniqueness.

Note:
`sDocName` does not support the use of localized document names. Legacy documents that use `sDocName` to reference the correct document name as stored in the CMS will continue to work. But it is recommended that you use the `iDocID` parameter instead.

Example:

```
http://<servername>:<port>/BOE/OpenDocument/opendoc/openDocument.jsp?sDocName=Sales+in+2003
```

5.2.3 sIDType

Syntax	Description	Values
sIDType	Specifies the type of object identifier used to specify the viewable document. Use in conjunction with iDocID.	• InfoObjectID • ParentID • CUID

Note:
If you pass in an InfoObject ID as a value to iDocID rather than a CUID, you do not need to specify the sIDType parameter. However, InfoObject IDs are changed when migrating documents from one CMS to another. It is recommended that the CUID be used, which is preserved during migration.

Example:

```
http://<servername>:<port>/BOE/OpenDocument/opendoc/openDocument.jsp?iDocID=Aa6GrrM79cRAmaOSMGoadKI&sIDType=CUID
```

5.2.4 sInstance

Syntax	Description	Values
sInstance	Specifies the scheduled instance of the target document to open. Use in conjunction with sDocName or iDocID.	• User (Latest instance owned by current user) • Last (Latest instance of the document) • Param (Latest instance of the document with matching parameter values. Crystal reports and Web Intelligence documents only.)

Example:

```
http://<servername>:<port>/BOE/OpenDocument/opendoc/openDocument.jsp?sDocName=Sales+in+2003&sInstance=User
```

5.3 Input parameters

5.3.1 lsC

Syntax	Description	Values
`lsC`	Specifies a contextual prompt if there is an ambiguity during SQL generation. **Note:** Only supported by Web Intelligence documents.	A prompt value that resolves the ambiguity in the SQL generation.

Example:

```
http://<servername>:<port>/BOE/OpenDocument/opendoc/openDocument.jsp?iDocID=Aa6GrrM79cRAmaOSMGoadKI&sIDType=CUID&lsC=Sales
```

5.3.2 lsI[NAME]

Syntax	Description	Values
`lsI`	Specifies index or key value. This parameter must be associated with one of the parameters lsS, lsM or lsR.	Value could be simple, multiple or a range according prompt type.

Example:

```
http://<servername>:<port>/BOE/OpenDocument/opendoc/openDocument.jsp?sDoc=IndexTest&sType=wid&lsMStore=[caption]&lsIStore=[index]
```

Example:

Using eFashion sample Universe passing a value for the "Store name" object which has been modified to be Index Aware as well as the Index Value for the "Store name" object:

```
http://localhost:8080/OpenDocument/opendoc/openDocument.jsp?sDoc=IndexTest&sType=wid&lsMStore=e-Fashion
New York Magnolia&lsIStore=2
```

5.3.3 lsM[NAME]

Syntax	Description	Values
`lsM[NAME]`	Specifies multiple values for a prompt. `[NAME]` is the text of the prompt.	• Multiple prompt values, separated by a comma. • `no_value` (only for optional parameters)

Note:
You can remove an optional parameter from the prompt by setting it to `no_value` in the openDocument query string. If you leave an optional parameter out of the openDocument query string, a default parameter value will be applied.

Example:

```
http://<servername>:<port>/BOE/OpenDocument/opendoc/openDocument.jsp?iDocID=Aa6GrrM79cRAmaOSMGoadKI&sID
Type=CUID&sRefresh=Y&lsMSelect+Cities=[Paris],[London]
```

```
http://<servername>:<port>/BOE/OpenDocument/opendoc/openDocument.jsp?iDocID=Aa6GrrM79cRAmaOSMGoadKI&sID
Type=CUID&sRefresh=Y&lsMparamStringDR=[c],[d]&lsMparamNumberDR=[3],[4]&lsMparamDate
DR=[Date(2003,6,3)],[Date(2003,6,4)]&lsMparamDateTimeDR=[DateTime(2003,6,1,3,1,1)],[DateTime(2003,6,1,4,1,1)]
```

Crystal reports

If the target is a Crystal report, `[NAME]` is the parameter name, and each parameter value must be enclosed in square brackets.

Example: Setting Crystal report parameters

```
http://<servername>:<port>/BOE/OpenDocument/openDocument.jsp?iDocID=ASsonFDFQtVOmHZZJJTJuSo&sIDType=CUID&lsM
SelectState=[Alberta],[Washington]
```

This example opens up a Crystal report with a parmeter named `SelectState` and sets its value to Alberta and Washington.

Web Intelligence documents

The character ? is a reserved prompt value for Web Intelligence documents in an openDocument URL. Setting the prompt value to lsM[NAME]=? in the URL forces the "Prompts" dialog box to appear for that particular prompt.

Olap Intelligence reports

If the target document is an OLAP Intelligence report (.car) you can use the IsM parameter to specify prompts. The parameters are passed in as a URL-encoded string using the unique name of the parameter set up in the OLAP Intelligence report.

Example: Setting a memberset parameter

```
http://<servername>:<port>/BOE/OpenDocument/openDocument.jsp?iDocID=544&sIDType=InfoObject&sType=car&ls
MADC216EA-D9A5-42B5-AE%2C21%2C84%2CA9%2CF9%2C6E%2C31%2C7=[%5BCustomers%5D.%5BCountry%5D.%26%5BMexi
co%5D],[%5BCustomers%5D.%5BCountry%5D.%26%5BCanada%5D]
```

This example opens up an OLAP Intelligence report with a memberset parameter to Customers > Country > Mexico and Customers > Country > Canada in the view.

5.3.4 lsR[NAME]

Syntax	Description	Values
lsR[NAME]	Specifies a range of values for a prompt. [NAME] is the text of the prompt. **Note:** Not supported by OLAP Intelligence reports.	• A range of values for the prompt, separated by a double period (..). • no_value (only for optional parameters)

Note:
You can remove an optional parameter from the prompt by setting it to no_value in the openDocument query string. If you leave an optional parameter out of the openDocument query string, a default parameter value will be applied.

Example:

```
http://<servername>:<port>/BOE/OpenDocument/opendoc/openDocument.jsp?iDocID=Aa6GrrM79cRAmaOSMGoadKI&sID
Type=CUID&sRefresh=Y&lsRTime+Period:=[2000..2004)
```

```
http://<servername>:<port>/BOE/OpenDocument/opendoc/openDocument.jsp?iDocID=Aa6GrrM79cRAmaOSMGoadKI&sID
Type=CUID&sRefresh=Y&lsRparamStringDR=[h..i]&lsRparamNumberDR=[7..8]&lsRparamCurrencyDR=[3..4]&lsRparamDat
eDR=[Date(2003,6,7)..Date(2003,6,8)]&lsRparamDateTimeDR=[DateTime(2003,6,1,7,1,1)..Date
Time(2003,6,1,8,1,1)]&lsRparamTimeDR=[Time(1,1,7)..Time(1,1,8)]&lsRparamUnbound1=(..6)&lsRpara
mUnbound2=[6..)&lsRparamStringR=[a..d]&lsRparamNumberR=[1..3]&lsRparamCurrencyR=[1..3]&lsRparam
```

```
DateR=[Date(2003,6,1)..Date(2003,6,3)]&lsRparamDateTimeR=[DateTime(2003,6,1,1,1,1)..Date
Time(2003,6,1,3,1,1)]&lsRparamTimeR=[Time(1,1,1)..Time(3,1,1)]
```

Crystal reports

If the target is a Crystal report, `[NAME]` is the parameter name, and the range must be enclosed in square brackets and/or parentheses (use a square bracket next to a value to include it in the range, and parentheses to exclude it).

5.3.5 lsS[NAME]

Syntax	Description	Values
`lsS[NAME]`	Specifies a value for a single prompt. `[NAME]` is the text of the prompt.	• A single prompt value. • `no_value` (only for optional parameters)

Note:
You can remove an optional parameter from the prompt by setting it to `no_value` in the OpenDocument URL. If you leave an optional parameter out of the OpenDocument URL, a default parameter value will be applied.

Example:

```
http://<servername>:<port>/BOE/OpenDocument/opendoc/openDocument.jsp?iDocID=Aa6GrrM79cRAmaOSMGoadKI&sID
Type=CUID&sRefresh=Y&lsSparamString=h&lsSparamNumber=1&
lsSparamCurrency=121&lsSparamDate=Date(2003,6,11)&
lsSparamDateTime=DateTime(2003,6,11,14,38,37)&lsSparamBoolean=false&
lsSparamTime=Time(12,39,2)&lsSparamStringDR=a&lsSparamDateDR=Date(2003,6,1)
```

Crystal reports

If the target is a Crystal report, `[NAME]` is the parameter name.

Example: Setting a Crystal report parameter

```
http://<servername>:<port>/BOE/OpenDocument/openDocument.jsp?iDocID=ASsonFDFQtVOmHZZJJTJuSo&sIDType=CUID&lsM
SelectState=California
```

This example opens up a Crystal report with a parmeter named `SelectState` and sets its value to California.

Web Intelligence documents

The character `?` is a reserved prompt value for Web Intelligence documents in an openDocument URL. Setting the prompt value to `lsS[NAME]=?` in the URL forces the "Prompts" dialog box to appear for that particular prompt.

OLAP Intelligence reports

If the target document is an OLAP Intelligence report (.car) you can use the lsS parameter to specify prompts. The parameters are passed in as a URL-encoded string using the unique name of the parameter set up in the OLAP Intelligence report.

Example: Opening an OLAP report to a specific page

If `23CAA3C1-8DBB-4CF3-BA%2CB8%2CD7%2CF0%2C68%2CEF%2C9C%2C6F` is the URL-encoded unique name for the page parameter in the OLAP Intelligence report, you would use the following URL to open the OLAP Intelligence report to page 2:

```
http://<servername>:<port>/BOE/OpenDocument/openDocument.jsp?iDocID=440&sIDType=InfoOb
ject&sType=car&lsS23CAA3C1-8DBB-4CF3-BA%2CB8%2CD7%2CF0%2C68%2CEF%2C9C%2C6F=2
```

Example: Setting a cube parameter

If `8401682C-9B1D-4850-8B%2C5E%2CD9%2C1F%2C20%2CF8%2C1%2C62` is the URL-encoded unique name for the cube parameter opening the warehouse cube in the catalogue FoodMart 2000 on MSAS, you would use the following URL to open this cube parameter:

```
http://<servername>:<port>/BOE/OpenDocument/openDocument.jsp?iDocID=616&sIDType=InfoOb
ject&sType=car&lsS8401682C-9B1D-4850-8B%2C5E%2CD9%2C1F%2C20%2CF8%2C1%2C62=CATALOG%3DFood
Mart%202000,CUBE%3Dwarehouse
```

5.3.6 sPartContext

Syntax	Description	Values
`sPartContext`	Specifies the data context of a report part. Use in conjunction with `sReportPart`. **Note:** Only supported by Crystal reports.	The name of the report part data context.

Note:
Only mandatory if a value is specified for `sReportPart`.

Example:

```
http://<servername>:<port>/BOE/OpenDocument/opendoc/openDocument.jsp?iDocID=Aa6GrrM79cRAmaOSMGoadKI&sIDType=CUID&sReportPart=Part1&sPartContext=0-4-0
```

Note:
The `sReportPart` and `sPartContext` parameters are only supported with the DHML parts viewer (`sViewer=part`).

5.3.7 sRefresh

Syntax	Description	Values
`sRefresh`	Indicates whether a database refresh should be forced when the target document is opened.	• `Y` • `N`

Certain documents can contain saved settings to specify that a database refresh must occur when the document is opened in a viewer. These document settings will override `sRefresh=N`.

Example:

```
http://<servername>:<port>/BOE/OpenDocument/opendoc/openDocument.jsp?iDocID=Aa6GrrM79cRAmaOSMGoadKI&sIDType=CUID&sRefresh=Y
```

Crystal reports

The `sRefresh` parameter is only supported with the `html` and `part` Crystal report viewers, and not the `actx` and `java` viewers.

5.3.8 sReportMode

Syntax	Description	Values
sReportMode	Indicates whether the link should open the full target Crystal report or just the report part specified in sReportPart. **Note:** Only supported by Crystal reports.	• Full • Part

Note:
Defaults to Full if this parameter is not specified. Only applies if a value is specified for sReportPart.

Example:

```
http://<servername>:<port>/BOE/OpenDocument/opendoc/openDocument.jsp?iDocID=Aa6GrrM79cRAmaOSMGoadKI&sIDType=CUID&sReportPart=Part1&sReportMode=Part
```

5.3.9 sReportName

Syntax	Description	Values
sReportName	Specifies the report to open if the target document contains multiple reports.	The report name for Web Intelligence documents and page name for OLAP Intelligence reports.

Note:
Defaults to the first report if this parameter is not specified.

Example:

```
http://<servername>:<port>/BOE/OpenDocument/opendoc/openDocument.jsp?iDocID=Aa6GrrM79cRAmaOSMGoadKI&sIDType=CUID&sReportName=First+Report+Tab
```

5.3.10 sReportPart

Syntax	Description	Values
sReportPart	Specifies the part of the target Crystal report to open. **Note:** Only supported by Crystal reports.	Name of the Crystal report part.

Example:

```
http://<servername>:<port>/BOE/OpenDocument/opendoc/openDocument.jsp?iDocID=Aa6GrrM79cRAmaOSMGoadKI&sIDType=CUID&sReportPart=Part1
```

Note:
The sReportPart and sPartContext parameters are only supported with the DHML parts viewer (sViewer=part).

5.4 Output parameters

5.4.1 NAII

Syntax	Description	Values
NAII	Indicates whether to force the display of the prompt selection page. **Note:** Only supported by Web Intelligence documents.	• Y (prompt values that are passed with lsS, lsM, or lsR in the URL are applied and not displayed in the "Prompts" dialog box)

Note:

- NAII=Y raises the "Prompts" dialog box for any values not specified in the URL. Prompts created with default values are still displayed in the "Prompts" dialog box.

- If all prompt values are specified in the URL, the prompt window does not appear even if `NAII=Y` is specified.

Example:

This example assumes there are two prompts in the Web Intelligence document: `Year` and `Country`. `NAII=Y` forces the "Prompts" dialog box to appear and allows the user to specify a value for the `Country` prompt. The `Year` prompt is already set to a value of `FY1999` in the URL using the `lsS` parameter and therefore is not prompted for.

```
http://<servername>:<port>/BOE/OpenDocument/opendoc/openDocument.jsp?iDocID=Aa6GrrM79cRAmaOSMGoadKI&sID
Type=CUID&lsSYear=FY1999&NAII=Y&sRefresh=Y
```

5.4.2 noDocument

Syntax	Description	Values
`noDocument`	A value of `true` forced a report to open in design mode using the existing report template. **Note:** Only supported by Web Intelligence documents.	Boolean value: `true`

Note:

- `noDocument=true` automatically forces a Web Intelligence report into design mode.
- Since the existing report template is used, you can prevent overwriting this template by applying the appropriate security.

Example:

```
o http://<server>:8080/BOE/OpenDocument/opendoc/openDocument.jsp?iDocID=6471&noDocument=true
```

5.4.3 sViewer

Syntax	Description	Values
`sViewer`	Specifies the selected report viewer.	• `html` • `part` (Crystal reports only)

Example:

```
http://<servername>:<port>/BOE/OpenDocument/opendoc/openDocument.jsp?iDocID=Aa6GrrM79cRAmaOSMGoadKI&sID
Type=CUID&sViewer=html
```

5.4.4 sOutputFormat

Syntax	Description	Values
`sOutputFormat`	Specifies the format in which to open the target document.	• `H` (HTML) • `P` (PDF) • `E` (Microsoft Excel (97-2003) - Crystal reports only) • `W` (Rich Text Format (RTF) - Crystal reports only)

Note:
Defaults to HTML if this parameter is not specified.

Example:

```
http://<servername>:<port>/BOE/OpenDocument/opendoc/openDocument.jsp?iDocID=Aa6GrrM79cRAmaOSMGoadKI&sID
Type=CUID&sOutputFormat=E
```

More Information

Information Resource	Location
SAP product information	http://www.sap.com
SAP Help Portal	http://help.sap.com/businessobjects Access the most up-to-date English documentation covering all SAP BusinessObjects products at the SAP Help Portal: • http://help.sap.com/bobi (Business Intelligence) • http://help.sap.com/boepm (Enterprise Performance Management) • http://help.sap.com/boeim (Enterprise Information Management) Certain guides linked to from the SAP Help Portal are stored on the SAP Service Marketplace. Customers with a maintenance agreement have an authorized user ID to access this site. To obtain an ID, contact your customer support representative. To find a comprehensive list of product documentation in all supported languages, visit:http://help.sap.com/boall.
SAP Support Portal	http://service.sap.com/bosap-support The SAP Support Portal contains information about Customer Support programs and services. It also has links to a wide range of technical information and downloads. Customers with a maintenance agreement have an authorized user ID to access this site. To obtain an ID, contact your customer support representative.
Developer resources	http://www.sdn.sap.com/irj/sdn/bi-sdk-dev https://www.sdn.sap.com/irj/sdn/businessobjects-sdklibrary
SAP BusinessObjects articles on the SAP Community Network	http://www.sdn.sap.com/irj/boc/articles These articles were formerly known as technical papers.

Information Resource	Location
Notes	https://service.sap.com/notes These notes were formerly known as Knowledge Base articles.
Forums on the SAP Community Network	https://www.sdn.sap.com/irj/scn/forums
Training	http://www.sap.com/services/education From traditional classroom learning to targeted e-learning seminars, we can offer a training package to suit your learning needs and preferred learning style.
Consulting	http://www.sap.com/services/bysubject/businessobjectsconsulting Consultants can accompany you from the initial analysis stage to the delivery of your deployment project. Expertise is available in topics such as relational and multidimensional databases, connectivity, database design tools, and customized embedding technology.

Index

D

document history 5

I

iDocID 21

L

logon tokens 14
lsC 24
lsM 25
lsR 26
lsS 27

M

migration 8

N

NAII 31
noDocument 32

P

parameters
  deprecated 9
  list of 17
  obsolete 9

S

sDocName 22
serialized sessions 13
serSes 19
session management 13
sIDType 22
sInstance 23
sOutputFormat 33
sPartContext 28
sRefresh 29
sReportMode 29
sReportName 30
sReportPart 31
sViewer 32
syntax 11

T

token 20

U

user sessions 15

W

what's new 8