Build Information - Build 111

From InfoBurst Support Wiki

Release Date

February 11, 2010

User Interface

Version

2.6.680

Excel Format Tab Splitting

The split of a document by report tab for publishing in Excel format is now supported.

Context

Context support added. If the document uses context, context is displayed in Document Properties and configurable in the burst under Parameter Sources.

User Migration

InfoBurst administrators can now migrate all objects owned by a user and user group memberships to another user account. Group migration is required so ensure the target user will be able to browse to the objects.

Platform

Version

2009.14.111

Minor Revisions

- Split and Combine is now supported for Excel format (and Static Excel)
- When a Document is re-cataloged, all prompts in bursts will be updated
- Delivery format @Format/@EDLFormat is supplied to all queries to allow format selection by recipient. Values are *PDF,EXCEL,HTML,TEXT,PNG* and *NATIVE*
- New Email options at Burst Level (EmailFrom, EmailFooter)
- Email TOC option revised so if disabled no content headers will be generated
- PNG delivery now defaults to all pages plus validation on page range
- Static PDF now supports PNG output (and page range selection)

Object References

Every Object in the system that has a reference to another object is now tracked and this means better integrity as it should not be possible to delete an object that has been used.

For example a burst will reference one or more documents and a database query could be used as a data source within a burst. Objects can be listed as those that reference an object or objects that are linked to an object.

Web Service APIs

GetObjectReferences(ByVal Token As String, ByVal ObjectType As ObjectType, ByVal ObjectID As Integer, ByVal SubObjectID As Integer, ByVal GetBasicHistory As Boolean) As IB_ObjectResult

GetLinkedObjects(ByVal Token As String, ByVal ObjectType As ObjectType, ByVal ObjectID As Integer, ByVal GetBasicHistory As Boolean) As IB_ObjectResult

Script Commands

refs object="x" [qry="x"]

links object="x"

Excel Post-Processing

It is now possible to provide two generic Excel functions that can be applied to a list of worksheets. The functions are built-in to InfoBurst 2009 and do not require Excel. Excel functions are currently available via the command shell interface.

- **AutoFilter**: Adds a drop-down to each column of the worksheet that allows for a quick selection
- **Group**: Group rows into an outline to make it easier to see large amounts of data. Each group will be collapsed and can be expanded when required. By default grouping will be done for the first column and a header row can be specified.

The implementation of these functions has been achieved by adding *extensions* internally to the *BurstDistribution* object and therefore additional extensions can be added in the future.

Web Service API

IB_BurstDelivery has been extended to contain an *extensions* collection of type **IB_BurstDeliveryExtension**.

To configure **AutoFilter**, create a new IB_BurstDeliveryExtension as follows:

- ExtensionType = ExcelPostProcess
- ExtensionFunction="autofilter"
- ExtensionParameter1="Worksheet(s)"

To configure **Group**:

- ExtensionType = ExcelPostProcess
- ExtensionFunction="group"
- ExtensionParameter1="Worksheet(s)"
- ExtensionParameter2="Column"
- ExtensionParameter3="bHeaders"

where

"WorkSheet(s)" is a string that lists one or more sheets separated with ","

"Column" is the column to Group. If blank "A" will be assumed "bHeaders" is a boolean indicating if a header row exists. Supply "true" or "false"

Script Commands

The *Add Delivery* command supports the following new keywords:

- xlaf
- xlgrp
- xlgrpcol
- xlgrphdr

Example 1: *Add Delivery ... xlaf="Sheet1,Sheet2"*

Example 2: *Add Delivery ... xlgrp="Sheet2" xlgrpcol="A" xlgrphdr="true"*

Delivery Conditions

It is now possible to add a condition to a delivery so that a specific delivery can be sent only when the condition is true.

If more than one condition exists, all must be True for the delivery to take place.

The condition is a free format *SQL*-like command that allows for some flexibility such as *"cond1 AND cond2"* or things like *inlist('a','b')*.

A simple condition could be *"State='Arizona'"* or a more complex condition like *"State inlist('Arizona','California')"*

Web Service APIs

IB_BurstDelivery has been extended to contain an *conditions* collection of type **IB_BurstDeliveryCondition**. Each IB_BurstDeliveryCondition has a string that is set to the condition.

Script Commands

The **Add Delivery** command now supports the condition keyword.

Example: *Add Delivery ... condition="State='Texas'"*

Migration APIs

To facilitate customers moving/upgrading their XI servers some new API functions are available that can be used to build new UI options / custom tools to allow for various scenarios where existing platforms and documents need to be migrated to new XI versions.

Web Service APIs

Document_Validate(ByVal Token As String, ByVal DocumentID As Integer) As IB_DocumentSummary

Document_Remap(ByVal Token As String, ByVal DocumentID As Integer, ByVal NewPlatformID As Integer) As IB_DocumentSummary

Platform_ChangeType(ByVal Token As String, ByVal PlatformID As Integer, ByVal NewType As ObjectSubType) As IB_DocumentSummaryResult

Platform_ValidateDocuments(ByVal Token As String, ByVal PlatformID As Integer) As IB_DocumentSummaryResult

Document_Validate will check that the current ID on the Platform is correct and if not will update it. This is useful if a customer has deleted and recreated a Document but used the same location and name.

Document_Remap will update all instances of the given Document to point to a new Platform and then update the ID.

Platform_ChangeType allows an existing Platform to become a different type (such as XIR2 becomes XI31). This is when customers upgrade their XI server and keep the CMS name the same. The call will then validate all Documents on the platform and update the IDs.

Platform_ValidateDocuments will check that the current ID for all Documents on the given Platform are correct and if not will update them.

Script Commands

validate doc id="x"

validate platform id="x"

Next build will add support for changing platform type and remapping documents.

REST Support

The REST interface has been significantly updated in this release as it will be required to support the new XDC/XDM Flex connectors and also we are starting to explore new ideas such as iPhone applications.

The idea behind REST is that it operates over HTTP using GET or POST requests and therefore almost ANY device or technology can call it. You can even test it by entering the URI into a browser.

In this build any call into the REST interface will cause a user/password challenge (if you use a browser you will see a dialog asking for your credentials).

For the ultimate in cross-platform interoperability this release also offers the choice of either XML or JSON for some calls and can also support POST for some calls also.

To call the REST interface, construct a URI that points to the InfoBurst2009 server like this:

http://myserver:8551/infoburst/rest

Then add the specific command such as *"/list/folders"*.

Here is a list of the supported REST functions:

/list/folders	List all Root folders plus User Root
/list/folder/**id**	List objects and sub-folders for a Folder
/list/dbconn/**id**	List queries for a DBConn
/search/**object**	Find objects (in repo, not folders)
/exec/dbqry/**id**?params=xx	Run a Query

JSON formatted output is available using the following commands:

/list.json/folders /list.json/folder/**id** /search.json/**object**

As the REST interface develops a REST Reference guide will be created to document in more detail each call and its return data.

XDC For Excel

XDC now supports manually defined cell ranges. In some cases an Excel workbook may be too complex or fragmented to automatically discover *parts* so a new attribute **XDSAutoParse** has been added to **IB_Document** that allows an Excel Document to be flagged as NOT compatible with the parts model and therefore manually defined cell ranges can be used.

When a Document is added to an XDC, the attribute is carried over in the **IB_XmlDataCacheSource** as **SupportsAutoParse**. To add a manually defined cell range, simply set the Source to "Sheet1@Range".

Example: *"Sheet1@A2:C10"*

Crystal XIR2/XI3x Database Logons Support added to supplying database logon credentials. IB_Document has been extended to include a new collection *Logons* of type **IB_DocumentLogon** that contains a list of required database logons for the document. This collection is created when the Crystal document is added to InfoBurst.

To manage the authentication to the database, a new set of DBLogons has been added to the User and a new set of API's allow the creation etc.

Web Service APIs

AddUserDBLogon(ByVal Token As String, ByVal UserID As Integer, ByVal Server As String, ByVal Database As String, ByVal User As String, ByVal Password As String) As Boolean

ModifyUserDBLogon(ByVal Token As String, ByVal UserID As Integer, ByVal Server As String, ByVal Database As String, ByVal User As String, ByVal Password As String) As Boolean

RemoveUserDBLogon(ByVal Token As String, ByVal UserID As Integer, ByVal Server As String, ByVal Database As String) As Boolean

Script Commands

add dblogon user="x" server="x" database="x" logonuser="x" password="x"

modify dblogon user="x" server="x" database="x" logonuser="x" password="x"

delete dblogon user="x" server="x" database="x"

Self-Hosted UI

As of this release the Platform will serve the UI directly without needing IIS.

Direct the browser to the following URL:

http://server:8550/infoburst

Support for XIR2 SP6 and XI3 SP2

Two new handlers have been added to sync with changes made by SAP/BusinessObjects to the web service consumer libraries.

The SOAP call Modify_Platform has been changed to allow a change of Type where the major version is the same so this allows XIR2 platforms to be changed to SP5 or SP6 and XI3 platforms to be changed to SP1 or SP2.

The Script command *modify platform* now supports the new keyword *type* to allow for the above change.

Example: *modify platform* xir2 *type=*xi26

IB2.6 XDS Compatibility Interface

The four XDS aspx scripts that are used for IB 2.6 XDS deployments are now available in a new self-hosted service and provide compatibility with existing XDS solutions.

By default, these new services are NOT started.

The following new configuration options have been added:

- IB26Mode
- IB26ServicePort
- IB26Folder

IB26Mode can be set to *Enabled* or *EnabledWinAuth* (the default is Disabled). The IBServer must berestarted after changing IB26Mode.

The *EnabledWinAuth* setting sets up the service to require Windows Authentication and would typically be used when using GetCredentials.

The **IB26ServicePort** (default 8553) controls the port # where the services are hosted.

The **IB26Folder** (default *\IB26*) is where DBConnections and XDC Objects should be located.

When the service is first enabled, the IB26Folder will be created if it does not exist (plus the DBConn and XDC sub-folders).

The service endpoints are hosted at http://server:8553/infoburst/services

- **GetXDS**
- **GetCredentials** (Get_Credentials in IB 2.6)
- **Writeback** (XC_Writeback in IB 2.6)
- **ExecQry** (XC_RTRead in IB 2.6)

The query string arguments are fully supported just like IB 2.6

If a dashboard currently calls

http://server/infoburst/services/getxds.aspx?k=MyCache

the new call would be

http://server:8553/infoburst/services/getxds?k=MyCache

Note the :8553 port and the removal of .aspx

Retrieved from "http://wiki.infosol.com:80/infoburst/index.php?title=Build_Information_-_Build_111&oldid=24"

- This page was last modified on 13 February 2014, at 18:24.
- This page has been accessed 548 times.