Web Intelligence Caching in BusinessObjects Enterprise XI Release 2

Applies to:

BusinessObjects Enterprise Web Intelligence XIR2

For more information, visit the Business Objects homepage.

Summary

This document explains how Web Intelligence caching works in BusinessObjects Enterprise XIR2. Understanding the caching behaviors will help in tuning the environment to be more efficient and can help in increasing performance of Web Intelligence reporting.

Author: Carly Trow, Senior Engineer

Company: SAP Business Objects

Created on: 03 December 2007

Author Bio

Carly Trow, Senior Engineer

I started with Business Objects in January 2005 on the Web Intelligence team. Within a year, I was collaborating with colleagues on training plans for WebI and even spent some time in India training newly hired engineers.

In October 2006, I accepted the role of a Web Intelligence Resource on the Architecture team, which has translated to Senior Engineer in the current support structure on the IDD-WebI team.

Table of Contents

Introduction

WebIntelligence does not use a separate cache server as Crystal Reports and Desktop Intelligence does. The WebIntelligence Report Server and the Business Objects SDK handles caching. The purpose of caching is to prevent the report from hitting the database again and process the report again, thus saving time and processing cycles.

WebIntelligence Workflows

The following workflows represent a single server machine process flow which may or may not include multiple WebI Report Servers. It is important to note that in a clustered environment the cache is associated to the Report Server that created it. Therefore if a request hits the Business Objects Enterprise system, and the architecture load balances the request to a WebI Report Server that does not contain the cached files, the performance gains resulting from the generated cache will not be realized.

With the introduction of SP2 to XI R2, the ability exists to share these cache files amongst WebI Report Servers – see section below, "WebIntelligence Cache Sharing".

Viewing on Demand

Scenario: User attempts to view a WebI report that has been recently added to a Public Folder and has not yet been viewed. No cache currently exists and the report has "Refresh on Open" disabled.

1. The WebI Report Server retrieves the report template file (.wid) and the associated Universe template file (.unw) from the Input File Repository's File Store
2. The WebI Report Server merges the report data into the template and copies the resulting .wid file to the cache directory
3. WebI's cache directory post CHF15:
   a. Unix: <BOInstallDir>/bobje/data/<**server_CMSport**>/storage/docs/<#>/#.wid_
   b. Windows: <BOInstallDir>\BusinessObjects Enterprise 11.5\Data\<**server**>\storage\docs\<#>\
   c. Ex:
      `/home/carly/xir2/bobje/data/kermit_50064/storage/docs/00000088/`
      `00000002.wid_/00000002.wid`

> **Note:** XML files are created to reproduce the structure, format, and design of the report. If any calculations are executed using the report data, the resulting calculations are included in the XML data. No further calculations are required once the XML files are created.

4. The XML files are generated in that cache directory under:
   a. In UNIX
      i. XIR2 though SP2: /en_US/XMLPaginated/<page#>/Blob#.xml
         - Full path example:
           `/home/carly/xir2/bobje/data/kermit_50064/storage/docs/000`
           `00088/00000002.wid_/en_US/XMLPaginated/1/Blob1.xml`
      ii. XIR2 SP3: /en_US/**XMLPaginated.drilloff**/<page#>/Blob#.xml
         - Full path example:
           `/home/carly/xir2/bobje/data/kermit_50064/storage/docs/000`
           `00088/00000002.wid_/en_US/XMLPaginated.drilloff/1/Blob1.x`
           `ml`
   b. In Windows
      i. XIR2 through SP2: \en_US\XMLPaginated\<page#>\Blob#.xml
         - Full path example:

- `E:\xir2\BusinessObjects Enterprise`
  `11.5\Data\carlyr2sp2\storage\docs\00000057\00000001.wid_`
  `\en_US\XMLPaginated\1\Blob1.xml`
  ii. XIR2 SP3: `\en_US\`**`XMLPaginated.drilloff`**`\<page#>\Blob#.xml`
    - Full Path example:
    - `E:\xir2\BusinessObjects Enterprise`
      `11.5\Data\carlyr2sp2\storage\docs\00000057\00000001.wid_`
      `\en_US\`**`XMLPaginated.drilloff`**`\1\Blob1.xml`
5. A Blob#.xml file is generated for each page and report of the WebI document.
6. The XML file is generated as each page is viewed.

   NOTE: any report level functions, variables, calculations are performed during the .wid to .xml file creation.
7. The XML file is stored in cache until the cache file size limit or cache duration time has been exceeded.
8. As a user views the report the XML files are converted into HTML files viewable by the end user's browser by the Business Objects SDK (REBEAN).
9. The conversion from XML to HTML is known as XSL Transformation and occurs on the client side. Extensible Style-sheet Language Transformations (XSLT) is an XML-based language used for the transformation of XML documents into other XML or "human-readable" documents. The original document is not changed; rather, a new document is created based on the content of an existing one. XSLT is most often used to convert data between different XML schemas or to convert XML data into HTML or XHTML documents for web pages, creating a dynamic web page, or into an intermediate XML format that can be converted to PDF documents.

Viewing a Scheduled Instance

The following scenarios assume all WebI Report and Job servers are on the same server machine or there is only one set of WebI Report and Job servers.

Scenario: User1 schedules a WebI Report to refresh using all of the default options of the scheduler (default formatting, default destination, no caching options set, etc). User1 logs into Infoview and views the WebI Report instance.
1. During scheduling, the WebI Report Server retrieves the report template and merges it with data refreshed from the reporting database.
2. A copy of the resulting .wid is saved to the FRS Output File Store.
3. When the report is viewed, the same scenario occurs above except that the .wid is retrieved from the Output FRS File Store location and copied into the WebI Report Server's cache directory.
4. The XML files are generated one by one as User1 views each page of the instance in the same location as when viewing the report on demand.

Scenario: User2 logs into Infoview and views the report instance that User1 had scheduled above.
1. By default, the cache generated by User1 will only be used if User1 and User2 belong to the same user group with the same permissions on the report. Therefore User2 will see the same initial delay as cache is being generated that User1 sees when viewing the instance for the first time.
2. This delay is caused as the files are being regenerated in the WebI Report Server temp session folder:
   a. `<$BOBJEDIR>/data/<server>/<server>.Web_IntelligenceRepor tServer/sessions/...`
3. This delay may be negligible depending on the report's complexity. If many report level variables exist that perform calculations on the report's data cube, this initial view time may take much longer.
4. To share user cache between users, enabling the following permission from the CMC at the report or folder level:

        a. Home > BusinessObjects Enterprise Applications > Web Intelligence > Advanced Rights : "Merge dimensions for synchronization"
        b. Enable this for either the user or user group. This will allow for User2 to be able to utilize User1's generated cache.

5. For very complex reports where the initial view of the instance takes too long to display, HTML pre-caching should be utilized during scheduling. See below for more information on this option.

NOTE: If Report and Job Servers are located on separate physical servers, cache sharing between these servers can be achieved by adding the command line switch –storageRoot <shared path> to the WebI Report Servers' command line. In UNIX this would be accomplished through the ccm.config file.

Report level functions, variables, calculations are performed during the .wid to .xml file creation.

WebIntelligence Cache Settings in the CMC
(Excerpts from the XIR2 Deployment and Configuration Guide)

List of Values Caching

Enables or disables caching per user session of list of values in WebIntelligence Report Server. The default is for the feature to be on.

Enable Viewing Caching

When this parameter is on, real-time caching is possible for WebIntelligence documents when they are viewed, or when they are generated as a result of having been run as a scheduled job.

When this parameter is off both real-time caching of WebIntelligence documents and viewing of cached Web Intelligence documents is impossible.

Real-time caching is done only if both this parameter and the Enable Real Time Caching parameters are on.

Enable Real Time Caching

When this parameter is on, it allows the WebIntelligence Report Server to load document states in active memory as they are viewed or run as scheduled jobs. This will increase performance of report navigation when viewing on demand because it stores the previously viewed pages in memory.

When the parameter is off, the Web Intelligence Report Server does not cache the Web Intelligence documents in memory, but cache files will continue to be written to the WebI Report Server's cache directories if Viewing Caching is enabled.

This parameter is taken into account only when the Enable Viewing Caching is set to on.

Note: To improve system performance, set the "Maximum Number of Downloaded Documents to Cache" to zero when this option is selected, but enter a value for Maximum Number Of Downloaded Documents To Cache when this option deselected.

Document Cache Duration

The amount of time (in minutes) that content is stored in cache.

Document Cache Size

The size (in kilobytes) of the document cache.

Amount of Cache to Keep When Document Cache is Full

If the storage size is bigger than the allocated storage size, the system will delete documents with the oldest "last accessed time." Then if the cache size still exceeds the maximum storage size, the Web Intelligence Report Server will clean up the cache until the amount of cache percentage is reached.

Document Cache Scan Interval

The number of minutes that the system waits before checking the document cache for cleanup.

Maximum Number of Downloaded Documents To Cache

The number of Web Intelligence documents that can be stored in cache.

Note:	To improve system performance, set this value to zero when 'Enable Real Time Caching' is selected, but enter a value when Enable Real Time Caching is deselected.

WebIntelligence Cache Sharing

In a clustered environment that spans multiple server machines, it is possible to configure XI R2 w/SP2 to share the cache files generated by the WebI Report Server. To do so you must add the following switch to the startup command line

```
-storageRoot <SharedDirectoryPath>
```

Where <SharedDirectoryPath> is a UNC Share in a Windows environment or a UNIX mount point that has 'Full Control' rights and permissions to the service or user account running the WebI Report Server service or daemon.

Note: In a Windows environment, it is inappropriate to use a drive letter mapping as it is specific to a user environment profile and would not exist when the code is running in 'service' mode

Scheduling Using Pre-cache Options

HTML Pre-Cache

HTML Pre-caching can be used to generate the XML files at schedule time instead of at view time of the instance. This is particularly useful for complex reports including those that contain many report level variables that perform calculations on the data stored in the data cube of the WebI document.

Using HTML pre-caching in WebIntelligence does not actually generate HTML files, but XML files. The XML files are converted into browser readable files (HTML, image files) when viewed. The SDK does this XSL Transformation on the fly.

To enable HTML pre-caching from the CMC:
1. Login to the CMC.
2. Navigate to the report to be scheduled.
3. Click Properties > Schedule > Caching Options
4. Under "Select the Formats to Pre-Load the Cache with when Scheduling" check "Standard HTML"
5. Next select the locale the cache files are to be created in.
6. Click the "Update" button.
7. Click back on the "Schedule" tab.
8. Click "Schedule" button.

To enable HTML pre-caching from Infoview:

1. Login to the Infoview.
2. Navigate to the Folder containing the report to be scheduled.
3. Click "Schedule" link below report name.
4. Expand the "Caching the Options" section.
5. Under "Select the Formats to Pre-Load the Cache with when Scheduling" check "Standard HTML"
6. Next select the locale the cache files are to be created in.
7. Click "Schedule" button.

When the report schedule runs, the same caching workflow occurs as mentioned in the workflows above. The extra step taken for pre-caching is that the .wid is converted into the XML files at schedule time prior to the user viewing the instance.

One XML file is generated per report page.

These XML files are stored in the following location:
*   XIR2 through SP2: /en_US/XMLPaginated/<page#>/Blob#.xml
*   XIR2 SP3: /en_US/XMLPaginated.drilloff/<page#>/Blob#.xml

Other XML files are cached at schedule time that are entire reports rather than being broken down 1 XML blob per page. This XML is used for viewing a WebI document in Draft Mode. It is cached under the WebI Report cache directory:
*   `/en_US/XMLDraft/#/Blob#.xml`

The "#" directory in this case is related to the reports or sub-reports within a WebI document (tabs) and not related to a page number.

Excel Pre-Cache

Excel pre-caching is similar to HTML pre-caching except that the report is converted to an Excel format at the time of scheduling and stored in cache.

This cache will be used when saving or exporting the WebI Report instance to Excel rather than having to perform the conversion on the fly.

The resulting XLS file is cached under the WebI Report Server cache directory:
*   Standard: /en_US/Excel/1/Blob#.xls
*   Optimized for Formatting: /en_US/ExcelOptimised/1/Blob#.xls
*   Full Path Example:
    `/home/carly/xir2/bobje/data/kermit_50064/storage/docs/00000030 /00000002.wid_/en_US/Excel/1/Blob3.xls`

PDF Pre-Cache

This cache will be used when viewing, saving or exporting the WebI Report instance in PDF format rather than converting on demand.

The resulting PDF file is cached under the WebI Report Server cache directory:
*   en_US/PDF/<page#>/Blob#.pdf
*   Full Path Example:
    `/home/carly/xir2/bobje/data/kermit_50064/storage/docs/00000030 /00000002.wid_/en en_US/PDF/1/Blob1.pdf`

Related Content

BusinessObjects Enterprise XIR2 Administrator's Guide:
http://help.sap.com/businessobject/product_guides/boexir2/en/xir2_bip_Admin_en.pdf

BusinessObjects Enterprise XIR2 Administrator's Reference Guide:
http://help.sap.com/businessobject/product_guides/boexir2/en/xir2_bip_Admin_en.pdf

Information on XSL Transformation:
http://en.wikipedia.org/wiki/XSL_Transformations

For more information, visit the Business Objects homepage.

Copyright