ICA Client Bandwidth Analysis Test Environment

By Citrix Consulting Services

Citrix Systems, Inc.

Notice

The information in this publication is subject to change without notice.

THIS PUBLICATION IS PROVIDED “AS IS” WITHOUT WARRANTIES OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. CITRIX SYSTEMS, INC. (“CITRIX”), SHALL NOT BE LIABLE FOR TECHNICAL OR EDITORIAL ERRORS OR OMISSIONS CONTAINED HEREIN, NOR FOR DIRECT, INCIDENTAL, CONSEQUENTIAL OR ANY OTHER DAMAGES RESULTING FROM THE FURNISHING, PERFORMANCE, OR USE OF THIS PUBLICATION, EVEN IF CITRIX HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES IN ADVANCE.

The exclusive warranty for Citrix products, if any, is stated in the product documentation accompanying such products. Citrix does not warrant products other than its own.

Version History		
December 5, 2001	Melissa S. Hopkins Melissa.Hopkins@Citrix.com	Version 1.0

Table of Contents

Document Overview

This document describes the testing environment used to perform bandwidth testing for the ICA Client Bandwidth Analysis white paper. The purpose of the ICA Client Bandwidth Analysis white paper is to describe the bandwidth impact of common ICA Client settings in a MetaFrame environment. The following settings were tested:

- **Use Data Compression**: Data compression reduces the amount of data that needs to be transferred but requires additional processor resources to compress and decompress the data.
- **Use disk Cache for bitmaps**: Bitmap caching to disk stores commonly used graphical objects such as bitmaps in a local disk cache on the client but uses local hard disk space.
- **Queue mouse movements and keystrokes**: Queuing causes the client to send mouse and keyboard updates less frequently to the Citrix server. Leaving this option unchecked makes the session more responsive to keyboard and mouse movements.
- **SpeedScreen Latency Reduction**: Latency reduction is a collective term used to describe the functionality that helps enhance user experience on slower network connections.
    - **Mouse Click Feedback**: Displays instant feedback for mouse clicks on the client device.
    - **Local Text Echo**: Accelerates the display of input text on the client device.

The above settings were configured into eleven different combinations to form the following test case scenarios:

ID	Use Data Compression	Use Disk Cache for Bitmaps	Queue Mouse Movements and Keystrokes	Speed Screen Latency Reduction	
				Mouse Click Feedback	Local Text Echo
Data_01	(No Settings Configured – Baseline Test Case)				
Data_02	X				
Data_03*	X	X			
Data_04	X	X			
Data_05	X		X		
Data_06	X	X	X		
Data_07**	X			X	X
Data_08	X			X	X
Data_09	X	X		X	X
Data_10	X		X	X	X
Data_11	X	X	X	X	X

*Data_03 is a duplicate of Data_04, but it is executed without a baseline cache for only the PowerPoint test (executing the typing test with Use Disk Cache for Bitmaps enabled does not produce any cache files).

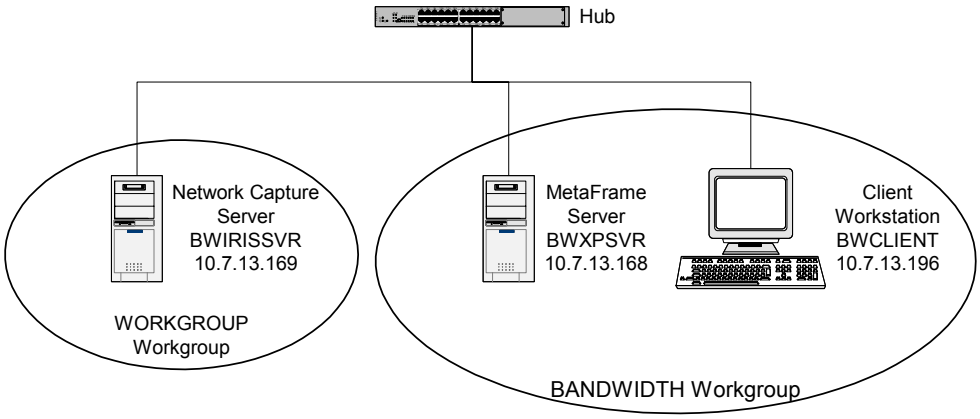
**Data_07 is a duplicate of Data_08, but it is executed without a baseline cache for both the typing test and the PowerPoint test.

This document describes:

- The test environment architecture for the ICA Client bandwidth analysis testing
- The test setup process including:
  - WinBatch script creation
  - ICA File configuration
  - Performance Monitor log setup
  - Baseline cache directory configuration
- The Iris configuration settings used to capture data
- The test execution process
- The data summarization process used to analyze ICA Client bandwidth data.

Test Environment Architecture

The ICA Client bandwidth test environment consisted of a MetaFrame server, a server used for network captures, and a client workstation, all on a single isolated network connected by a hub. A hub was used in this testing environment in order to watch all ICA traffic between the client workstation and the MetaFrame server from a separate network capture server (a switch may be used if it has port mirroring capabilities). A separate network capture server was used to avoid placing additional load on either the client workstation or the MetaFrame server. A diagram of the testing environment is shown below, followed by the detailed specifications of each component.



Hub Specifications

- 3Com OfficeConnect Hub 8/TPO (Product #: 3C16700)
- 8 ports
- 10Mb Ethernet

Network Capture Server Specifications (BWIRISSVR)

- Compaq Proliant DL360
- Dual Intel Pentium III 800Mhz processors
- 1GB RAM
- 2 mirrored 9.1 GB hard drives
  - Hidden System Partition – 36MB
  - C: Drive – 4GB
  - D: Drive – 4.5GB
- Network
  - 2 Compaq NC3163 Fast Ethernet NIC adapters, only the first is enabled (supports promiscuous mode)
  - No DHCP
  - IP Address: 10.7.13.169
  - Subnet Mask: 255.255.255.0

- Software
  - Windows 2000 Server, SP2
  - Eeye Iris 3.60 Build 3 (latest evaluation version from www.eeye.com)
- Display Configuration
  - 1024x768
  - True Color (24 bit)

MetaFrame Server Specifications (BWXPSVR)

- Compaq Proliant DL360
- Dual Intel Pentium III 800Mhz processors
- 1GB RAM
- 2 mirrored 9.1 GB hard drives
  - Hidden System Partition – 36MB
  - C: Drive – 4GB
  - D: Drive – 4.5GB
- Network
  - 2 Compaq NC3163 Fast Ethernet NIC adapters, only the first is enabled
  - No DHCP
  - IP Address: 10.7.13.168
  - Subnet Mask: 255.255.255.0
- Software
  - Windows 2000 Server, SP2
  - MetaFrame XP 1.0, FR1 (local MS Access Data Store, farm name - BWTEST)
  - Citrix Resource Management Services (RMS) 1.0b
  - Microsoft Office 2000
  - Adobe Acrobat Reader 5.0.1
  - WinZip 8.0
- Display Configuration
  - 1024x768
  - True Color (24 bit)
- Published Applications: "MS Word", "MS PowerPoint"
  - 1024X768
  - True Color (24 bit)
  - Maximize application at startup
  - Audio On
  - Basic Encryption

- Normal CPU Priority
- Local Administrators and Local Users are permitted access

- User Configuration
  - Local username: bwuser, Password: password

Client Workstation Specifications (BWCLIENT)

- Dell Optiplex G1 (Service Tag 3PI2B)
- Intel Celeron 433 Mhz processor
- 128MB RAM
- 6GB hard drive
  - C: Drive – 4GB
  - D: Drive – 2GB
- Network
  - 3Com 3C918 Integrated Fast Ethernet Controller (3C905B-TX Compatible)
  - No DHCP
  - IP Address: 10.7.13.196
  - Subnet Mask: 255.255.255.0
- Software
  - Windows 2000 Professional, SP2
  - ICA Client V 6.20.985
  - WinBatch 2001N (latest evaluation version from www.winbatch.com)

Test Setup

Once the Test Environment was configured as described in the previous section, the following initial setup tasks were performed:

1. Two WinBatch scripts were created for a typing test and a PowerPoint test.
2. Custom ICA files were created based on the ICA Client settings required for each test case.
3. A Performance Monitor Log file was configured on the client machine.
4. Baseline disk cache directories were created for disk cache and local text echo settings.
5. Iris configuration settings were modified and documented.

These tasks are described in further detail in the sections that follow.

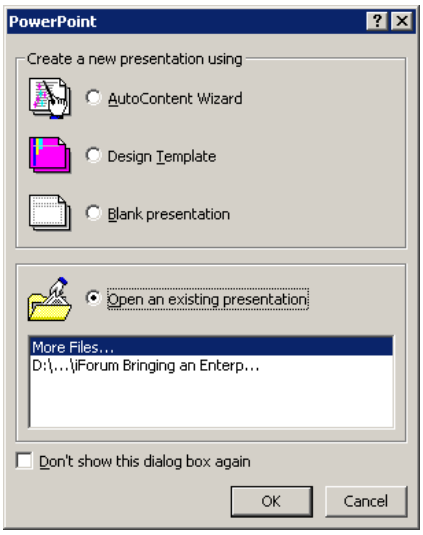
WinBatch Scripts

Two WinBatch scripts were created for the bandwidth analysis testing, a typing test script (TypeTest.wbt) and a PowerPoint test script (PowerPointTest.wbt). Both scripts were created as a combination of coded Windows Interface Language (WIL) functions and recorded keystrokes/mouse movements using WinBatch's WinMacro recording utility. An evaluation copy of WinBatch 2001N, obtained from www.winbatch.com, was used to create the scripts.

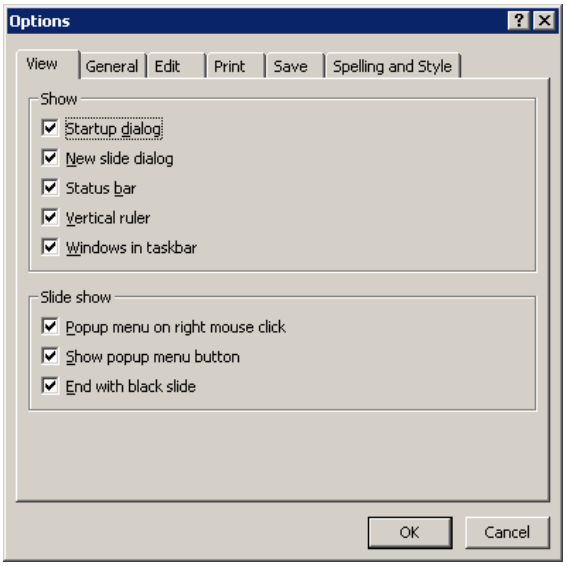
The TypeTest.wbt script was originally created by recording a Word typing session. However, when the recording is translated into a wbt file, the timing between each typed character is summed up and placed at the end of several characters, causing the script to send several keys to the screen and then wait approximately 3-4 seconds. To create a more accurate depiction of a user typing at 40-50 words per minute, a wbt file was coded using WIL functions instead of using the WinMacro translation. The coded script opens the Word.ica file using the Run function and creates ten paragraphs of six sentences each, with ten words in each sentence ("1Words 2Words", etc.). A short time delay was included after each character to more closely simulate a typing speed of 40-50 words per minute. The script varies the font type several times and saves to D:\Typing Test.doc after every three or four paragraphs using keyboard shortcuts.

The PowerPointTest.wbt script was created using the WinMacro recorder and then translated into a wbt file. The script first opens PowerPoint, maximized to take up the entire desktop. The script then opens the first PowerPoint presentation file in the D:\Presentation directory on the MetaFrame server using keyboard shortcuts. A slide show is then started using the keyboard shortcut F5, and the mouse is used to click through each slide. When the slide show is over, the mouse is used to first close the presentation and then the PowerPoint application by clicking on the appropriate X's in the top right corner of the session. The wbt file was slightly modified to use the Run function to execute the PowerPoint.ica file.

Note: The PowerPointTest.wbt script requires that the PowerPoint Startup Dialog is enabled and appears when starting PowerPoint. For clarification, the Startup Dialog is shown below:



To ensure that the Startup Dialog is enabled, ensure that the Startup Dialog checkbox is checked under Tools/Options as shown below:



The completed WinBatch scripts (TypeTest.wbt and PowerPointTest.wbt) were copied to the root of the D: drive on the client workstation and were executed from this location during testing.

ICA Files

The WinBatch scripts each launch a published application session on the MetaFrame server by executing an ICA file on the client. The TypeTest.wbt script launches a “MS Word” published application by running the Word.ica file, and the PowerPointTest.wbt script launches a “MS PowerPoint” published application by running the PowerPoint.ica file.

The Word.ica and PowerPoint.ica files were created in two steps. First, a template ICA file was generated for each published application as follows:

- Open the CMC on the MetaFrame server.

- Expand the applications node.
- Right-click on the published application.
- Click **Create ICA File** from the published application context menu.
- Choose **Minimal explanation required** and click **Next** in the first screen of the ICA File Wizard.
- In the ICA File Settings (1 of 2) screen, verify the following settings:
    - **Compress ICA data stream** is checked.
    - Session Window Size is set to **Absolute (in pixels)** with a Width of 1024 and Height of 768.
    - **Use Seamless if Available** is checked.
    - Colors are set to **True Color (24 bit)**.
    - Encryption is set to **Basic**.

  Click **Next**.
- In the ICA File Settings (2 of 2) screen, verify the following settings:
    - **Use TCP/IP + HTTP Browsing** check box is checked.
    - Audio Level is set to **Low sound quality**.
    - The default path is C:\Program Files\Citrix\Administration and the ICA File Name is `Word.ica` or `PowerPoint.ica` depending on the published application.

  Click **Next**.
- In the Create HTML File screen, choose **No** when prompted to create an HTML template for this application. Then click **Next**.
- Click **Finish** in the ICA File Summary screen to generate the ICA file template.

Once the template ICA files were generated, the ICA files were modified to include additional parameters to configure the ICA client settings being tested. The following table describes these parameters:

ICA Client Setting	ICA file Parameter	Description	Value	Default Value
Use Data Compression	Compress	Configures data compression	Enabled = On Disabled = Off	On
Use Disk Cache for Bitmaps	PersistentCacheEnabled	Enables Disk Caching	Enabled = On Disabled = Off	Off for LAN* On for WAN*
	PersistentCacheSize	Sets the maximum disk space in bytes allowed for caching bitmaps	10485760 (10 MB)	10MB
	PersistentCacheMinBitmap	Defines the minimum size in bytes a bitmap must be before it will be cached	8192 (8KB)	8KB
	PersistentCachePath	Specifies the directory path to the disk cache	D:\ICAClient\Cache	%UserProfile%\ Application Data\ ICAClient\Cache
Queue Mouse Movements and Keystrokes	MouseTimer	Time interval in milliseconds during which mouse input is collected before being sent to the MetaFrame server	Enabled = 100 Disabled = 0	0 for LAN* 100 for WAN*

ICA Client Setting	ICA file Parameter	Description	Value	Default Value
	KeyboardTimer	Time interval in milliseconds during which keyboard input is collected before being sent to the MetaFrame server	Enabled = 50 Disabled = 0	0 for LAN* 50 for WAN*
Mouse Click Feedback	ZLMouseMode	Configures the Mouse Click Feedback setting	Enabled = 1 Disabled = 0 Auto = 2 (not used in bandwidth tests)	0
Local Text Echo	ZLKeyboardMode	Configures the Local Text Echo setting	Enabled = 1 Disabled = 0 Auto = 2 (not used in bandwidth tests)	0

*The default values for these settings vary based on the Connection Type specified in the ICA Client. LAN in the table above refers to the Local Area Network connection type; WAN refers to the Wide Area Network connection type.

One additional parameter, MaximumCompression, was added to the ICA files. This parameter controls the size of the buffer used by the data compression process. When enabled, the buffer is maximized to produce larger and therefore most likely fewer, ICA packets. This parameter is automatically enabled when Use Disk Cache for Bitmaps is enabled. Therefore, to ensure that the same packet size is used whether or not Use Disk Cache for Bitmaps is enabled, this parameter was enabled (set to On) for all tests.

The following is an example of the modified ICA file structure with the added parameters as listed above:

```
[WFClient]
Version=2
TcpBrowserAddress=10.7.13.168
HttpBrowserAddress=bwxpsvr

[ApplicationServers]
MS Word=

[MS Word]
Address=MS Word
InitialProgram=#MS Word
ClientAudio=On
AudioBandwidthLimit=2
TWIMode=On
DesiredHRES=1024
DesiredVRES=768
DesiredColor=8
TransportDriver=TCP/IP
WinStationDriver=ICA 3.0
BrowserProtocol=HTTPonTCP
;Ensure that the Reducer buffer size is maximized for all tests
```

```
MaximumCompression=On
;Use Data Compression
Compress=Off
;Use Disk Cache For Bitmaps
PersistentCacheEnabled=Off
PersistentCacheSize=10485760
PersistentCacheMinBitmap=8192
PersistentCachePath=D:\ICAClient\Cache
;Queue Mouse Movements and Keystrokes
MouseTimer=0
KeyboardTimer=0
;SSLR Mouse Click Feedback
ZLMouseMode=0
;SSLR Local Text Echo
ZLKeyboardMode=0
```

A separate ICA file was configured for each test and placed in the D:\ICA Files directory on the client workstation. The typing test ICA files were named Word – Data_xx.ica where xx is the number of test. The PowerPoint test ICA files were named PowerPoint – Data_xx.ica where xx is the number of test. Before each test was executed, the appropriate ICA file was copied to the D: root drive of the client workstation and renamed to either Word.ica or PowerPoint.ica accordingly.

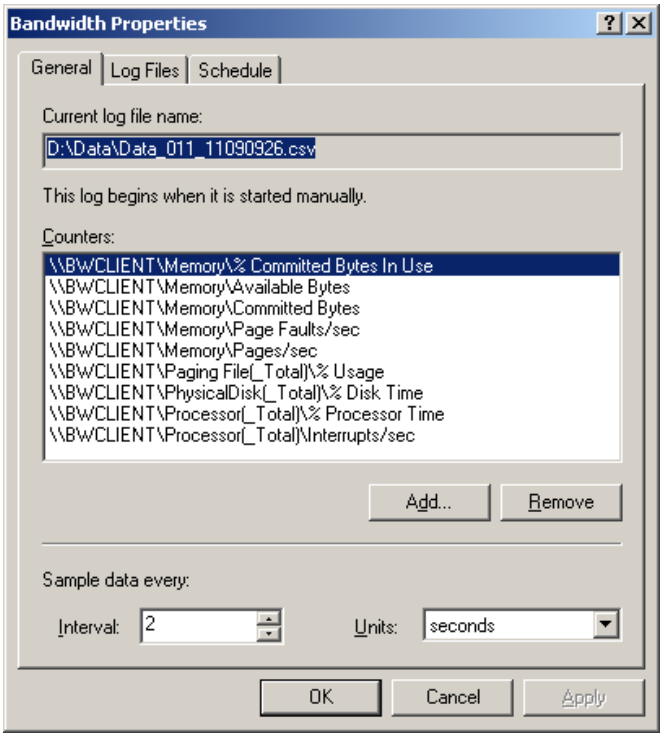
All ICA files were configured to use D:\ICAClient\Cache as the cache directory. Therefore, the following directories were also created on the client workstation to store cached files:

- D:\ICAClient\Cache (for the Use Disk Cache for Bitmaps setting)
- D:\ICAClient\Cache\zlcache (for the Local Text Echo setting)

Performance Monitor Logs

A Performance Monitor log was created on the client workstation during each test to monitor several basic counters including CPU and memory usage. The Performance Monitor counter log was created as follows:

1. From the Start menu, choose Programs, Administrative Tools, Performance.
2. Expand Performance Logs and Alerts in the left pane.
3. Click on Counter Logs to display all existing counter logs.
4. Right-click in the right pane, and click New Log Settings….
5. Enter Bandwidth as the name of the new log.
6. In the General tab, add counters as follows:



7. In the Log Files tab, change the location of the log files to D:\Data on the client machine. Enter an appropriate file name, for example Data_011 for the first iteration of Data_01.
8. On the Schedule tab, modify the start log settings to start the log manually.

A new log file was created for each test by stopping the log, changing the log file name, and restarting the log.

Baseline Cache Directories

Benefits of ICA Client caching functionality (Use Disk Cache for Bitmaps and Local Text Echo) are not realized until after a session has been initiated at least once and a baseline cache has been created. Once a baseline cache has been created, the next time a session is initiated that performs the same functionality (for example reviewing the same PowerPoint presentation or typing in the same font), the cache files are accessed locally instead of the same graphic files or font information being retrieved again from the server.

Baseline cache directories were created for the Use Disk Cache for Bitmaps and Local Text Echo settings by executing tests Data_03 and Data_07 for both the typing test and the PowerPoint test, and saving off the resulting cache files from each test to temporary baseline cache directories on the client workstation as follows:

- Data_03 typing test results saved to D:\TypingTestCache.
- Data_07 typing test results saved to D:\TypingTestZLCache.
- Data_03 PowerPoint test results saved to D:\PowerPointTestCache.
- Data_07 PowerPoint test results saved to D:\PowerPointTestZLCache.

While each test was executed three times, the cache directories were saved only once.

Before executing all tests, the ICA Client cache directories (D:\ICAClient\Cache and D:\ICAClient\Cache\zlcache) must be cleaned out. Before a test is performed that enables either Use Disk Cache for Bitmaps or Local Text Echo (with the exception of Data_03 and Data_07 as previously stated),

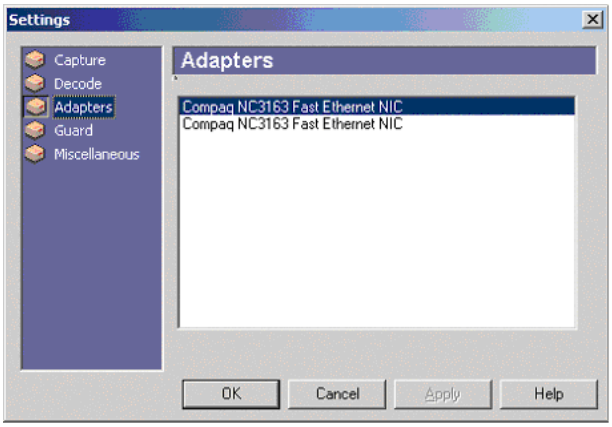
the appropriate temporary baseline cache directory must be copied into D:\ICAClient\Cache or D:\ICAClient\Cache\zlcache. Instructions for copying cache directories for each test are documented below in Section 0 Test Execution.

Iris Configuration Settings

The following sections describe the Iris configuration settings used for the ICA Client bandwidth testing. The evaluation version of Iris (version 3.6) used for testing was obtained from www.eeye.com.

Adapter Selection

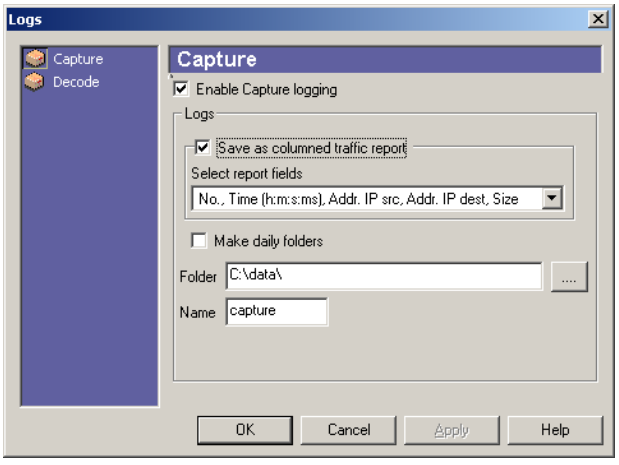
The first time Iris is started, the Adapter settings window appears and a network adapter must be selected.



The network capture server has two network adapters, however only the first adapter is enabled and attached to the network. Therefore, the first network adapter was chosen.

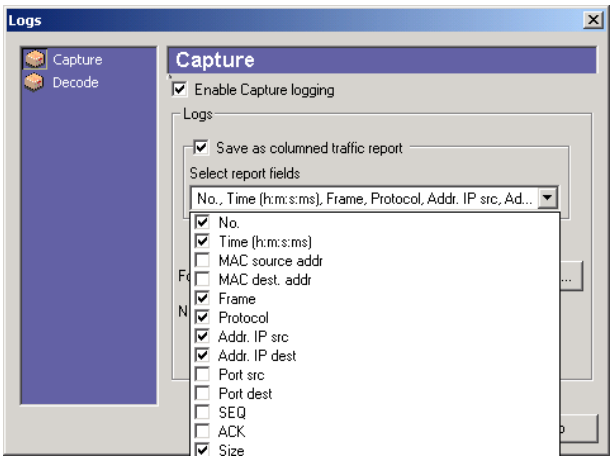
Capture Log Configuration

Capture logging must be enabled to save information about the captured packets during each test. In addition, the capture log settings must be configured to save the required information. Capture logging was enabled by choosing **Logs→Capture Logs** from the **Tools** menu and checking the **Enable Capture logging** check box.



The folder location was specified as D:\Data. And the name of the log file was changed before each test as described in step 2a of *Section 0 Network Capture Server Preparation*.

The capture logs were saved as a columned text file by checking **Save as columned traffic report** so that they could later be imported into a SQL Server database.



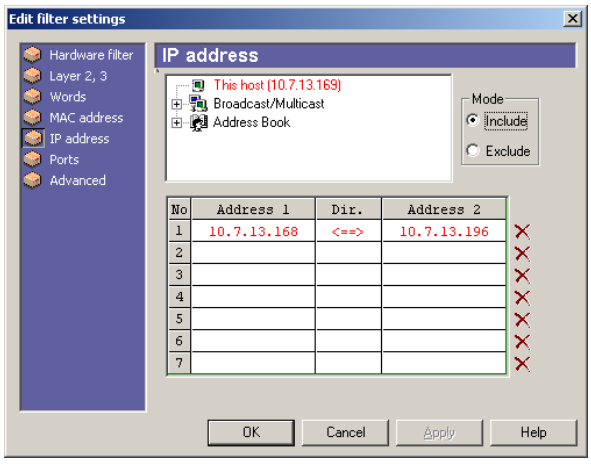
The following reports fields were chosen:

- No.
- Time (h:m:s:ms)
- Addr. IP src
- Addr. IP dest
- Size

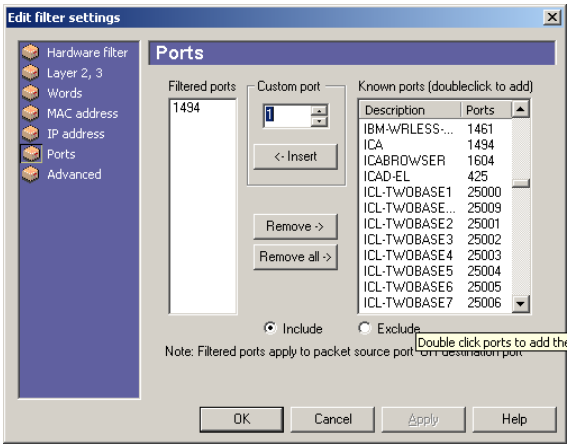
Configuring a Capture Filter

A capture filter was configured to specifically capture ICA packets between a client and server by choosing **Edit filter** from the **Filters** menu and creating the following filters:

- **IP address** was selected on the left pane in the window. **Include** was selected and a rule was configured to capture all packets between the IP address of the MetaFrame server and the IP address of the client workstation as shown below.



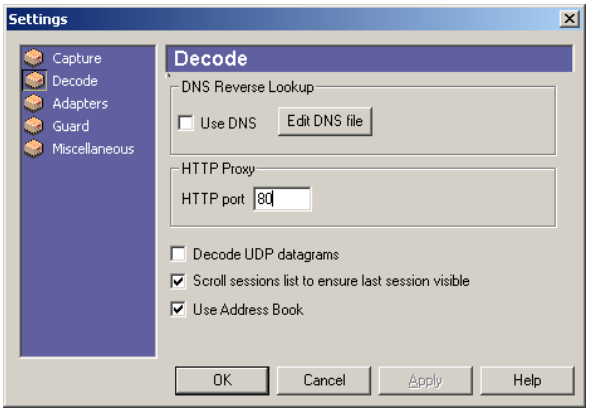
- **Ports** was selected on the left pane in the window. Include was selected and a rule was configured to filter on port 1494 (ICA) as shown below.



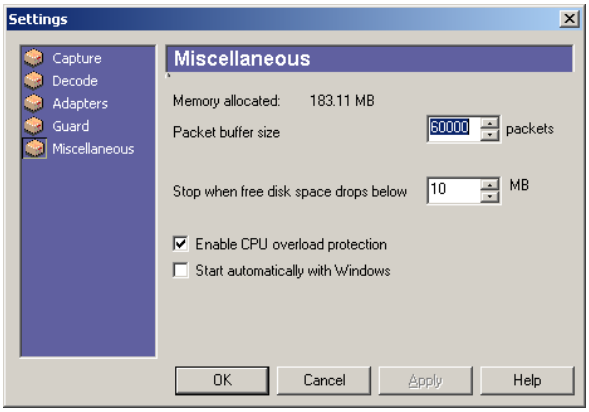
Additional Iris Settings

Additional configuration settings are available in Iris by choosing **Settings** from the **Tools** menu. The following options were configured:

- Since all tests were performed on an isolated network without DNS, Iris was configured to not use DNS by selecting the Decode option in the left pane of the window and uncheck the **Use DNS** check box.



- By default, Iris is configured with a packet buffer size of 2000 packets. When this buffer size is reached and capture logs are enabled, Iris will save the current buffer to a file and start a new buffer and file. Depending on the number of packets to be captured, the captured data may be broken into many files. To avoid this, the packet buffer size was set to its maximum by selecting Miscellaneous in the left pane of the window and entering 60,000 as the new packet buffer size.



Test Execution

Before each test is executed, the test environment must be prepared to capture the required data. The sections below describe the preparation process for each machine and the steps necessary to execute each test.

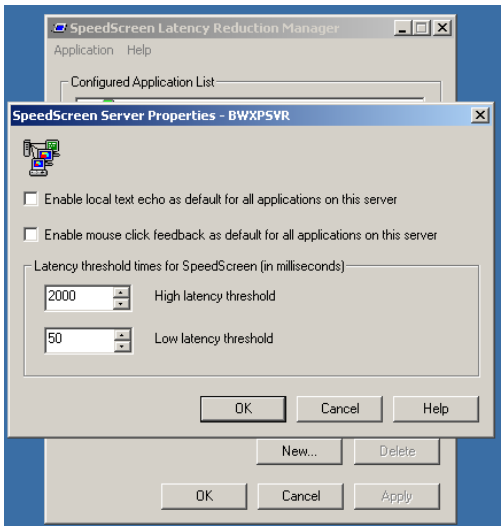
Client Workstation Preparation

1. Verify that the ICA file (`Word.ica` for the typing test, `PowerPoint.ica` for the PowerPoint test) is in the root of the D: drive and the ICA file settings are correct. Separate ICA files can be created for each test (`Word – Data_01.ica`, `Word – Data_02.ica`, etc.). If this is the case, copy the appropriate file to D: and rename accordingly.
2. Clear out the cache directories D:\ICAClient\Cache and D:\ICAClient\Cache\zlcache.
3. Copy the appropriate baseline cache directories as follows:
   a. Data_04, Data_06 – Copy the files from D:\TypingTestCache or D:\PowerPointTestCache to D:\ICAClient\Cache.
   b. Data_08, Data_10 – Copy the files from D:\TypingTestZLCache or D:\PowerPointTestZLCache to D:\ICAClient\Cache\zlcache.
   c. Data_09, Data_11 – Copy the files from D:\TypingTestCache or D:\PowerPointTestCache to D:\ICAClient\Cache. Also copy the files from D:\TypingTestZLCache or D:\PowerPointTestZLCache to D:\ICAClient\Cache\zlcache.

   NOTE: No cache files are generated during the typing test; therefore D:\TypingTestCache will be empty. Both the typing test and PowerPoint test generate zlcache files.
4. Change the name of the Performance Monitor log file to Data_*xxn* where *xx* is the number of the test and *n* is the test iteration (each test was executed 3 times).
5. Reboot the client workstation machine.

MetaFrame Server Preparation

1. For typing tests, delete the document D:\Typing Test.doc. For PowerPoint tests, verify that the presentation is located in the D:\Presentation directory.
2. Open the SpeedScreen Latency Reduction Manager utility and verify the server settings by double-clicking on the server name. The following window will appear:



For tests Data_01 through Data_06, verify that the **Enable local text echo as default for all applications on this server** and **Enable mouse click feedback as default for all applications on this server** settings are **unchecked**.

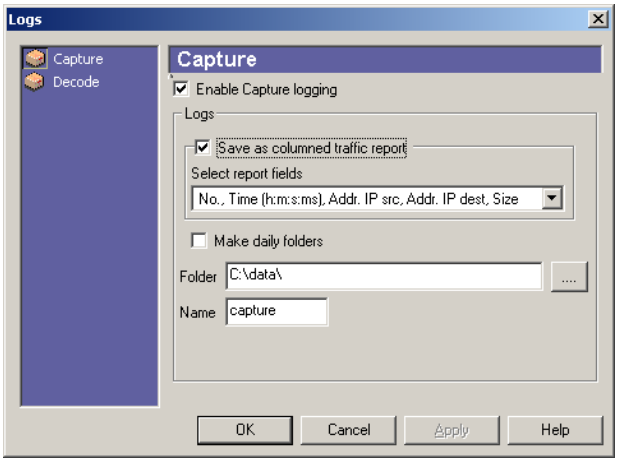
For tests Data_07 through Data_11, verify that the **Enable local text echo as default for all applications on this server** and **Enable mouse click feedback as default for all applications on this server** settings are **checked**.

3. Log out of the MetaFrame server.

Network Capture Server Preparation

In Iris, perform the following:

1. Start a new capturing session by choosing **File**|**New Capturing Session** or by clicking on the new capturing session icon to clear the displayed packets in the capture window.
2. Review the log file configuration by choosing **Tools**|**Logs→Capture Logs**.



a. Verify that **Enable Capture logging** is checked.

b. Verify the folder location for the logs.

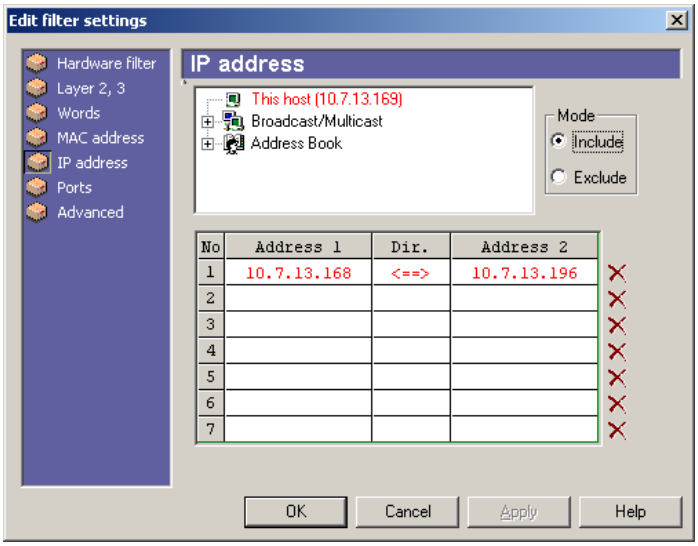
c. Modify the name of the log file to be Data_*xxn* where *xx* is the number of the test and *n* is the test iteration.

d. Verify that the reports fields selected are:

No., Time (h:m:s:ms), Addr. IP src, Addr. IP dest, Size.

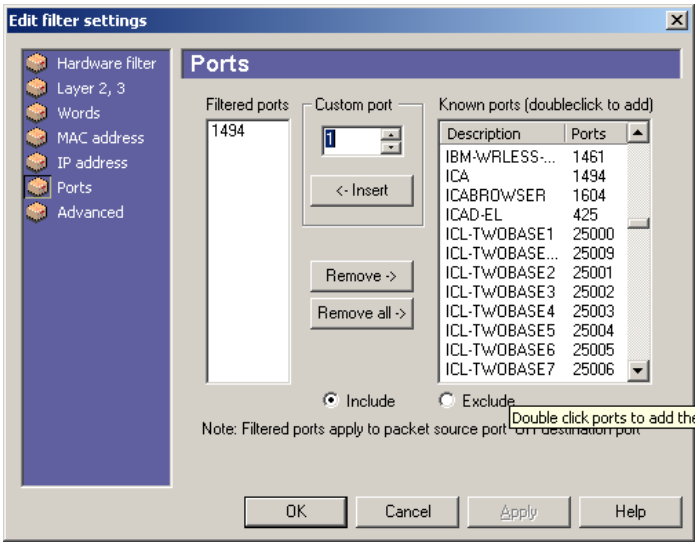
3. Verify the capturing filter in Iris by choosing **Filters**|**Edit filter**.

a. Select IP address on the left pane in the window.



Verify that **Include** is selected and that a rule is configured to capture all packets between 10.7.13.168 (MetaFrame server) and 10.7.13.196 (client workstation).

b. Select Ports on the left pane in the window.



Verify that **Include** is selected and that a rule is configured to filter on port 1494 (ICA).

Test Execution

The following steps were performed for each test:

1. Log in to the client workstation after the reboot.
2. Open an Explorer window on the client workstation and display the contents of the D: root directory. The following files should be visible:
    - `PowerPointTest.wbt`
    - `TypeTest.wbt`
    - `PowerPoint.ica`
    - `Word.ica`
3. Open Performance Monitor on the client workstation and expand the Counter Logs to display the Bandwidth counter log.
4. Start the capture on the network capture server.
5. Start the Bandwidth counter log on the client workstation and then minimize the Performance Monitor window.
6. On the client workstation, bring focus to the Explorer window and highlight the script to be executed.
7. Move the mouse to the bottom right-hand corner of the desktop, off the screen.
8. Start the WinBatch script by pressing Enter on the keyboard. If the WinBatch license agreement appears when starting a typing test, use the mouse to click agree, then quickly move the mouse back to the bottom right-hand corner of the desktop. If the WinBatch license agreement appears when starting a PowerPoint test, the focus of the published application may be on the PowerPoint main application window instead of the PowerPoint Startup Dialog once the PowerPoint application is opened. If this is the case, the script will not run correctly and must be restarted.
9. When the WinBatch script is finished, stop the Performance Monitor log on the client, and then stop the capture log in Iris to save the file.

Data Summarization

After all test results have been captured, the test result data must be accumulated in a standardized format that allows for further analysis. This section describes the process of standardizing time intervals for the Iris log file data, the steps performed to summarize the log file data, and the process of analyzing Performance Monitor processor utilization data. In addition, a description of tables used for Iris log file data summarization is included.

All SQL scripts, baseline table data files, and summarization spreadsheets are included in the following zip file:


Analysis.zip

Explanation of Standardized Time Intervals

Summarization of the Iris log file raw data was required before an analysis could be performed. All raw bandwidth data from the Iris log files were not initially summarized to a common time interval because Iris does not write data to the log file for every millisecond of time, only those milliseconds when packets are being transferred. Therefore, to analyze data based on a common time interval, all Size data (utilized bandwidth) was totaled for every second of data capture. To better understand the process used, an example of the actual layout of a log file should be examined:

Timestamp	Size
10:34:36:187	336
10:34:36:203	74
10:34:36:218	196
10:34:36:234	354
…	…
11:25:23:145	22
11:25:23:256	135
11:25:23:456	234

In this example, the data above is summarized into one-second intervals as follows:

Timestamp	Size
10:34:36	960
…	…
10:49:40	391

Every second of bandwidth information is then given an ID value beginning with 1, that corresponds to a one-second interval beginning with 1 second, as follows:

ID	Size
1	960
…	…
904	391

By beginning the time interval for each test at 1 second, data from each test case scenario can then be compared to all other test case scenarios to determine which configuration of ICA client settings provides the most optimal bandwidth usage.

To further prepare the data for analysis, a 15-second moving average was calculated before the data was graphed. This moving average allows for a weighted smoothing of the statistics, lessening the impact of bandwidth "bursts" on the final analysis.

Iris Log File Data Summarization

The following steps were performed to create the final bandwidth analysis spreadsheets based on the raw data from the Iris log files:

1. Execute the `1 - Create Bandwidth DB and Tables.sql` script on a Microsoft SQL Server 2000 database server. This script creates a database called "Bandwidth" and all the tables required to complete the additional steps listed below.
2. Use the SQL Server Data Transformation Services Import/Export Wizard to import all bandwidth data into the corresponding table. For example, the Typing Test data file for the first iteration of Data_01 should be imported into the bwTypeData011 table, for the second iteration of Data_01 should be implemented into the bwTypeData012 table, and so on.
3. Use the SQL Server Data Import Wizard to import data into the bwSecondsTable and the bwTimeTable.
4. Execute the `2 - Create TestSummary Data.sql` script to create the data for the bwTestSummary table. This table provides test summary statistics for each test iteration for both the Typing tests and the PowerPoint tests.
5. Execute the `3 - Create TestSummaryAvg Data.sql` script to average the test iteration summary data from the bwTestSummary table together to create one test summary row for each test in the bwTestSummaryAvg table. For example, test summary data for typing test iterations TypeData_011, TypeData_012, and TypeData_013 are averaged together to create one row of TypeData_01 data.
6. Execute the `4 - Create PP Sec Detail Data.sql` and `5 - Create Type Sec Detail Data.sql` scripts to standardize data from each detail table at the second time interval. For example, all data from the bwTypeData011 table will be summarized to the second level and placed in the bwTypeData011Sec table.
7. Execute the `6 - Create Sec Test Summary Data.sql` script to pull all data summarized at the second for each type of test into a single table. All bwTypeDataxxxSec tables are combined into one bwTypeTestSecSummary table, and all bwPPDataxxxSec tables are combined into one bwPPTestSecSummary table.
8. Execute `7 - Create PP Sec Test Summary Avg Data.sql` to create a moving average summary of the bwPPTestSecSummary table into the bwPPTestSecSummaryAvg table.
9. Execute `8 - Create Type Sec Test Summary Avg Data.sql` to create a moving average summary of the bwTypeTestSecSummary table into the bwTypeTestSecSummaryAvg table.
10. Export the data from the bwTestSummaryAvg, bwTypeTestSecSummaryAvg, and bwPPTestSecSummaryAvg tables into three text files using a comma separated value (CSV) format.
11. Import the CSV files into separate Excel spreadsheets.
12. Format the spreadsheet containing the bwTestSummaryAvg table for easy readability (to be copied into the white paper).
13. Use Excel's graphing capabilities to graph the data in the spreadsheets containing the bwTypeTestSecSummaryAvg and bwPPTestSecSummaryAvg tables. Graph the following data together:
    a. Data_011, Data_012, and Data_013 to determine if any test iteration does not follow the other two. Choose the test iteration that most follows an average path for use in the below graphs.
    b. Data_01 and Data_02.
    c. Data_03 and Data_04.
    d. Data_04, Data_05, and Data_06.
    e. Data_07 and Data_08.
    f. Data_02 and Data_08.
    g. Data_06 and Data_11.
    h. Data_09, Data_10, and Data_11.
14. Analyze the results from the graphs and the bwTestSummaryAvg spreadsheet to recommend optimal ICA Client configurations.

Data Summarization Tables

The following table describes the database tables created in Microsoft SQL Server 2000 used for summarization of the Iris log file data.

Table Name	Description
bwTimeTable	Used to summarize log file data at the second level. Contains one column of data, the time in seconds from 7:00:00 to 19:33:25. This table must cover every second interval during which bandwidth was captured.
bwSecondsTable	Used to standardize log file data to the same time interval, starting with 1 second. Contains two columns of data, an integer version of seconds used as an ID value from 1 to 945, and a char version of seconds used for comparison purposes to time values stored in char format from "001" to "945".
bwTempTimeTable	Temporary table used to store the start time and end time of a test iteration while standardizing the data at the second level. A "permanent" temporary table was required so that that temporary data would be available in the necessary context while executing the `4 - Create Sec Detail Data.sql` script.
bwTempTable	Temporary table used to store total bandwidth and total time information while creating rows in the bwTestSummary table. A "permanent" temporary table was required so that that temporary data would be available in the necessary context while executing the `2 - Create TestSummary Data.sql` script.
bwTestSummary	Stores summarized data calculated for each test iteration from the bwPPData*xxx* and bwTypeData*xxx* tables.
bwTestSummaryAvg	Test iteration summary data for each test stored in the bwTestSummary table was averaged together to create one row of averaged summary data for each test.
bwPPTestSecSummary	Data is read from all bwPPData*xxx* tables, standardized to the second level starting at 1 second, and then stored in this table as a summary. The table contains one column of bandwidth data for each test iteration (33 columns of bandwidth data).
bwPPTestSecSummaryAvg	A 15-second moving average is created from the data in bwPPTestSecSummary and stored in this table.
bwTypeTestSecSummary	Data is read from all bwTypeData*xxx* tables, standardized to the second level starting at 1 second, and then stored in this table as a summary. The table contains one column of bandwidth data for each test iteration (33 columns of bandwidth data).
bwTypeTestSecSummaryAvg	A 15-second moving average is created from the data in bwTypeTestSecSummary and stored in this table.
bwTypeData011 – bwTypeData113	These 33 tables store raw Iris log file data for each Typing test iteration (11 tests, 3 iterations for each test).
bwPPData011 – bwPPData113	These 33 tables store raw Iris log file data for each PowerPoint test iteration (11 tests, 3 iterations for each test).

Processor Utilization Data Summarization

During test execution, % processor utilization and several other metrics were captured on the client workstation using Performance Monitor logging. Data points were captured every two seconds and upon completion of each test iteration were saved as comma-separated value (CSV) files that can be read by Microsoft Excel.

Data compression functionality is designed to use additional processor resources on both the client workstation and the MetaFrame server. To quantify the increase in processor utilization when using data compression, Microsoft Excel was used to compare % process utilization data for tests Data_01 (without data compression) and Data_02 (with data compression). A Microsoft Excel worksheet was used to first consolidate % process utilization data for the Data_01 and Data_02 tests and standardize the data across the same two-second time intervals. A cross-section of the worksheet is shown below:

Seconds	% CPU Data_011	% CPU Data_012	% CPU Data_013	% CPU Data_021	% CPU Data_022	% CPU Data_023
1	100.00	100.00	100.00	100.00	100.00	100.00
3	7.50	7.00	6.50	5.50	7.00	6.50
5	15.50	20.50	3.50	6.00	15.50	5.00
7	19.50	30.00	18.00	18.50	54.00	22.50
9	2.00	78.00	69.50	86.50	83.00	78.50
11	97.50	100.00	59.50	21.50	17.50	53.50
13	43.00	100.00	11.50	59.13	1.00	8.00
...	...	...	...	...	...	...
671	4.50	0.00	2.00	14.00	0.00	0.50
673	0.00	9.00	0.00	2.00	0.00	0.00
675	0.00	0.00	0.00	6.00	0.00	0.00

Averages were taken of all test results and compared to determine the % increase in processor utilization. A second worksheet was used to calculate that moving average of the data so that the data could be graphed. These results are included in the final white paper.

6400 NW 6th Way **Fort Lauderdale, FL 33309** **954-267-3000** **http://www.citrix.com**