Wireless E-Mail Efficiency Assessment

RIM BlackBerry and Microsoft Direct Push

October 13, 2006

Table of Contents

1 Overview

An increasing number of users are taking advantage of wireless e-mail, which has become the most successful wireless-data application for wireless networks. Being able to respond to e-mail in real time makes workers more productive, recovers time that would otherwise not be available for productive use, and allows companies to serve their customers better. Increasingly, consumers also are using wireless e-mail—as pricing plans become less expensive and e-mail-capable devices such as smartphones and feature phones become more affordable.

With wireless e-mail rapidly reaching the point where it serves tens of millions of subscribers—and before long, hundreds of millions of subscribers—it is important to consider the effect that wireless e-mail systems have on wireless networks, especially given the limited data capacity of these networks. It is equally important to consider the end-user experience of these systems.

Under sponsorship from Research In Motion (RIM), Rysavy Research, working in conjunction with Quality in Motion, Inc., conducted a series of tests to quantify the data consumption of RIM's BlackBerry solution versus Microsoft's Direct Push solution. Direct Push requires Service Pack 2 for Microsoft Exchange Server 2003 on the server and Windows Mobile 5 on the device in combination with the Messaging and Security Feature Pack (MSFP).

The tests, as described in detail below, demonstrate that RIM's solution is significantly more efficient in its network usage, resulting in a much lower network load for an operator. The efficiencies also translate to a better user experience, with e-mails and attachments available much faster as well as improved battery life. This paper discusses the nature of wireless e-mail, wireless network capacity, how both Microsoft Direct Push and RIM BlackBerry work, the testing environment, the tests performed, and the test results.

2 Wireless E-Mail Systems

Wireless e-mail systems have become successful because they adapt an application intended for wireline networks to wireless networks. Traditional e-mail systems were designed using a model where e-mail clients poll e-mail servers for e-mail. Although it is possible to use such an approach with wireless connections, for handheld devices, this approach consumes too much data, is too slow, and does not result in an optimum user experience.

An optimized wireless e-mail systems provides the following types of features:

- **Push.** Rather than e-mail clients polling for data, the system "pushes" messages to the device.
- **Efficient.** Since some messages can be very large, the system makes it possible for users to read just the first part of the message rather than download the whole message. The user can then obtain additional portions of the message as desired.
- **Fast.** The system compresses the data to minimize communication time.
- **Efficient Attachment Handling.** Users typically want to view attachments, rather than manipulate them. The most efficient approach

lets the user view the attachment portion by portion and in a format optimized for the display on the device.

- **Mailbox Synchronization.** Users should be able to work with just one "mailbox," so that the handheld inbox is synchronized with the desktop or laptop inbox.
- **Security.** The system encrypts communication on an end-to-end basis and does not depend on any security provisions in the wireless network.
- **Sophisticated Management Tools.** The wireless e-mail system should provide tools for easily managing, updating, and deleting user accounts.

It is significant that the approaches used to minimize network communication also result in the best user experience. The less data sent over the wireless network, the faster and more reliable the user experience, and the better the device's battery life.

What became apparent in our testing is that there are very different implementation approaches for wireless e-mail and that the efficiency of these approaches can vary widely. One reason the efficiency of the wireless e-mail system is of particular concern to wireless operators is the limited capacity of their networks.

3 Finite Wireless Capacity

Rysavy Research has published extensively on the capacity of wireless networks, including a series of publicly available white papers for 3G Americas[1] as well as publicly available technology reports published through Datacomm Research, among them "Hard Numbers and Experts' Insights on Migration to Evolved 3G and 4G Wireless Technology."

Networks are becoming much faster than ever before. With technologies such as High Speed Downlink Packet Access (HSDPA) and CDMA2000 EV-DO (Evolution Data Optimized), users experience download throughput rates as high as 1 megabit per second (Mbps). However, what many people do not realize is that the capacity of these networks is relatively finite. Even though the wireless technologies use sophisticated approaches to maximize spectral efficiency, laws of physics impose a limit on the maximum number of bits per second that can be communicated in a radio channel.

Today's most advanced systems achieve a spectral efficiency of about .45 bps/Hz/sector.[2] This means that a system such as UMTS/HSDPA operating in a 5 MHz radio channel with half the capacity used for voice communications and half for data communications would have roughly 5 MHz divided by 2 times .45 bps/Hz, equaling 1.1 Mbps of sector throughput. EV-DO networks have similar constraints, with similar spectral efficiency, and typically only one radio channel of 1.25 MHz available for high-speed data.

Consider also that a voice channel consumes from about 6 to 12 kilobits per second (kbps) of capacity, depending on the exact vocoder and wireless network technology. A user operating at 1 Mbps, however, consumes one hundred times the capacity of a voice user while receiving data. Even though data usage tends to be sporadic, users can consume far more capacity with data communications than they can with voice communications. In round numbers, a voice user with a 1,000-minute plan can consume about 100 megabytes (MB) of data in a month. However, a data user can consume this

[1] http://www.3gamericas.com.
[2] 3G Americas white paper, "Data Capabilities: GPRS to HSDPA and Beyond", September, 2005.

much data downloading one 10-minute high-resolution video clip. For this reason, operators are sensitive to the types of applications their users are engaging in as well as how much capacity they need to provision per user. This is especially the case because many operators globally are not deploying 3G yet and because those that are typically only partially upgrade their networks to 3G. Hence, 2G and 2.5G networks with their more finite data capacity will be with us for many years.

To preserve wireless capacity, it is clearly important that applications operate as efficiently as possible. A wireless e-mail system that consumes much less data than another wireless system is going to have much lower impact on overall network data loading, thereby leaving greater spectral resources available for other users. This becomes ever more important as the number of wireless e-mail users keeps rising. More efficient wireless e-mail translates to a better user experience for all users and provides for a greater return on investment in wireless infrastructure and spectrum.

4 Overview of Testing

To test the efficiency of different wireless e-mail systems, Rysavy Research developed a test system that exactly replicates operation over a wireless network, but in a completely controlled environment. Rather than use an actual wireless network, the test system uses an advanced wireless network emulator. Communications with the handheld device occurred over a wireless connection provided by the network emulator, with all protocols identical to those used by a commercial network. The network emulator also allowed us to capture the data traffic, which we then analyzed for precise measurements of data communicated across the wireless interface.

We used controlled test messages of different types, varying both the size and the content. We designed the messages and attachments to be representative of what users might receive in a real-world situation. This paper describes the test messages and test apparatus in detail in subsequent sections.

We also configured the mail servers to represent real-world usage, and further configured the servers to communicate with our test wireless network using the exact protocols of commercial wireless networks.

We repeated each test five times. Given a high degree of consistency in the measurements, we achieved a high level of confidence in the test results, which are summarized in the next section.

5 Summary of Test Results

The following table summarizes the test results. The first column indicates the message size in bytes, the second column the type of attachment if any, the third column the size of the attachment in bytes, and the fourth column the combined size of the message plus attachment. Subsequent columns show the results for Direct Push and BlackBerry, listing the total number of bytes communicated over the radio interface as well as what percentage that number of bytes constituted relative to the size of the message (plus attachment if any).

A percentage value greater than 100 percent means that the wireless e-mail system communicated more bytes than the original message size, whereas a percentage value less than 100 percent means the wireless e-mail system communicated fewer bytes than the original message size. Lower percentage values represent better wireless efficiency.

Table 1: Summary of Test Results

				Direct Push		BlackBerry	
Message Size	**Attachment Type**	**Attachment Size**	**Message Plus Attachment**	**Sent Over The Air**	**% Sent**	**Sent Over The Air**	**% Sent**
5120	None	0	5120	12489	244%	3244	63%
10240	None	0	10240	16154	158%	6012	59%
20480	None	0	20480	20994	103%	11518	56%
5120	JPEG	152576	157696	270385	171%	11170	7%
5120	PDF text	165888	171008	251416	147%	187641	110%
5120	PDF image	108052	113172	183658	162%	26896	24%
5120	Word Doc	512000	517120	602027	116%	58209	11%
5120	PPT file	966656	971776	1450367	149%	266949	27%
5120	Excel	76800	81920	29055	35%	7581	9%

Notes: Sizes in bytes. Sent over the air includes both downstream and upstream transmissions. For messages/attachments available in portions, the test engineer requested "More" until the device received the entire message and attachment if any.

Appendix C of this paper provides actual test results. It shows the results of each individual test run as well as the data communicated in both the downstream and upstream paths.

Observations about the test results include:

- In all cases, BlackBerry was significantly more efficient than Direct Push.
- In nearly all instances, BlackBerry sends less data over the air than the original file size. In some cases, the amount sent is only a small percentage of the original file.
- In nearly all instances, Direct Push consumed more network resources than the original file size.
- One method by which BlackBerry achieves gains in network efficiency is by having efficient file viewers. In contrast, Direct Push downloads the whole file before the user can view it. On handheld devices, users generally want to view attachments rather than actually manipulate the files.

The more highly optimized data communications employed by BlackBerry have a number of important benefits. First, since it communicates less data, e-mail messages arrive faster. Given widely varying signal propagation environments, especially when mobile, this can mean the difference between receiving a message now or much later, in the event that a device loses the wireless signal for a period of time. Second, on any usage-based pricing plan, the user can send or receive many more messages for the same dollar amount. Third, the operator obtains a significant benefit by having a much lower load on its network. Finally, the user achieves better device battery life. The bottom line is a much more effective use of network resources with BlackBerry and a significantly better overall experience for the user.

6 Conclusion

Cellular operators are beginning to experience excellent success with wireless data services. In fact, data for many operators is exceeding 10 percent of their revenues. Wireless e-mail is one of the leading applications, with significant acceptance by businesses and now increasing adoption by consumers. While millions of people are using wireless e-mail today, the potential is for tens of millions in the next several years, growing eventually to hundreds of millions.

Operators are also promoting data service for other applications, such as mobile office productivity applications, Web browsing, and video and music downloads. With some data applications easily consuming an order of magnitude more network resources than voice communications, network capacity is an important consideration. Given the relatively low capacity of wireless networks—even 3G networks—compared to wireline networks, it makes excellent sense to emphasize applications that are efficient in their use of available bandwidth.

This white paper describes a set of wireless efficiency tests conducted by Rysavy Research that compare Microsoft Direct Push with RIM BlackBerry. These tests were done in a controlled test environment that exactly replicates operation over a wireless network. In every test performed, BlackBerry was significantly more efficient than Direct Push. In most instances, BlackBerry actually conserved bandwidth while Direct Push consumed more bandwidth than the original message.

The efficiency of RIM's solution benefits not only operators but also customers who consume a smaller amount of data, thus reducing monthly costs on some pricing plans, improving the speed with which users receive e-mail and attachments, and improving device battery life.

7 Appendix A: Wireless System Overview

Although Direct Push and BlackBerry provide a similar service to the end user, the overall architecture and system implementations are very different. This section describes how the two e-mail systems work as well as how they differ from each other.

7.1 Microsoft Direct Push

The Microsoft implementation has two main system components. First is the Microsoft Exchange Server, which is primarily responsible for sending and receiving e-mail messages. This requires Service Pack 2 running on Microsoft Exchange Server 2003. The second component is a device running Windows Mobile 5 in combination with the Messaging and Security Feature Pack (MSFP).

Once Direct Push is enabled on a client, Exchange will attempt to communicate with the respective device and synchronize messages on the server with the handheld. This solution requires the device to maintain a TCP connection with the server, accomplished by using a combination of heartbeat notifications and sustained TCP connections. Initially, the device sends a notification request to the server. This request contains a heartbeat interval, which is simply the maximum amount of time the device will wait until sending another notification request. If new mail arrives on the Exchange Server before the heartbeat expiration, Exchange notifies the device over the active connection.

The device will in turn request the new mail. If no new mail arrives before the heartbeat expires, the device will simply send the server another notification request to let it know that it is still waiting. All mail synchronization occurs over HTTP. This means that if the administrator fails to enforce HTTPS, e-mail is sent and received in the clear. Some wireless networks encrypt the radio link, but the encryption only protects a portion of the total end-to-end connection.

Microsoft Direct Push makes good use of standard networking protocols such as HTTP and TCP. However, Direct Push at this time is not nearly as optimized for a wireless environment as BlackBerry, which employs protocols specifically designed to withstand the variations experienced in wireless connections and to minimize the amount of data communicated.

7.2 BlackBerry

The RIM system and protocols are implemented in a much different manner. One important element of the system is the BlackBerry Enterprise Server (BES). This component consists of several Windows Services, which maintain the synchronization state of the system and communicate directly with the Exchange Server as well as with the RIM registration and data servers at the RIM Network Operations Center (NOC).

When the Exchange Server receives a message, it notifies the BES via the Messaging Application Protocol Interface (MAPI) subsystem. The BES then communicates the message to the RIM NOC, which in turn notifies the device of the pending message by sending a UDP message. The device and the RIM NOC then communicate to confirm availability. The BES then processes the message and sends it via UDP-based protocols to the device.

The BlackBerry messaging protocols are highly optimized to minimize the amount of communication across the wireless network, compared to the HTTP- and TCP-based approach used by Microsoft Direct Push. For instance, Direct Push transfers 12,591 bytes for a 5 KB[3] message; the BlackBerry solution transfers only 3,244 bytes for a 5 KB message.

It is also important to note that TCP itself is not an optimum protocol for wireless connections. For example, TCP algorithms can misinterpret a poor signal connection as a congested network. This results in inappropriate responses such as engaging the Slow Start algorithm, which further slows down communications. In contrast, BlackBerry protocols include an optimized transport protocol.

8 Appendix B: Test Environment

This section provides a detailed description of the testing environment, including the overall testing approach, the test messages, message sending approach, data analysis used, the Microsoft Direct Push test environment, and the RIM BlackBerry test environment.

8.1 Overall Testing Approach

Although the two systems Rysavy Research tested differed substantially in overall architecture and implementation, we were able to use a common testing environment to test each system. The primary test equipment was an Agilent 8960, a highly sophisticated wireless test system. This equipment combines a radio interface with a General Packet Radio Service (GPRS) Serving GPRS Support Node (SGSN) and Gateway GPRS Support Node (GGSN). In other words, it emulates an entire cellular operator network. The wireless device under test cannot differentiate between this and a commercial operator network. The Agilent equipment is able to capture the data traffic and make the IP traffic carried over the GPRS protocols available for analysis.

GPRS supports IP networking and provides a representative wireless environment for both Microsoft Direct Push and RIM BlackBerry. Though there are faster networks available, such as EDGE and EV-DO, the protocols used at the IP level for the wireless e-mail systems are the same. Essentially, the differences in the wireless technologies are at layers 1 and 2, so measuring the efficiency of the protocol at layer 3 (IP) over the wireless link provides a true "apples to apples" comparison of the wireless e-mail systems.

Both wireless systems involve communication with a Microsoft Exchange Server. The server hosted the accounts to which we sent test messages. We configured the Exchange Server to either send messages directly to the device in the case of testing Microsoft Direct Push or allow interaction with the BlackBerry Enterprise Server.

The next element of the test architecture was an Ethereal capture server. Ethereal, as described further in the next section, actively captured—via the Agilent equipment—all the data traffic being sent to and from the handheld device.

The final element was an ASP.NET test message creation and sending component. This component was an ASP.NET Web page written specifically to send the messages in the test matrix to the Exchange Server. The next section describes this process in greater detail.

[3] In this document, 1 KB = 1024 bytes.

8.2 Test Messages

To test the efficiency of the two systems, the test plan used a range of messages to approximate data usage in a real-world scenario. The messages had a range of body sizes as well as a variety of attachment sizes and types. The following table shows the test messages used.

Table 2: Test Message Types

Message Body Size	Attachment Size	Attachment Type
5 KB	0 KB	N/A
10 KB	0 KB	N/A
20 KB	0 KB	N/A
5 KB	149 KB	JPEG
5 KB	162 KB	PDF Text only
5 KB	106 KB	PDF Text + Images
5 KB	500 KB	Word doc
5 KB	944 KB	PowerPoint
5 KB	75 KB	Excel Spreadsheet

The first three test cases sent only a message without an attachment. This represents the majority of e-mails users receive. The test engineers created the contents of these messages by selecting random segments from a variety of news articles. This form of text represents the text used in typical e-mails and enabled the wireless e-mail systems to engage their compression schemes.

The next six cases all added an attachment of various size and type. The test engineers chose attachments that represent the types of documents sent by a typical office worker. These include a JPEG image file, two types of PDF documents (one text and one with an image), a Microsoft Word document, a Microsoft PowerPoint file, and a Microsoft Excel file. When sending attachments, the test plan called for a consistent and small message body to determine how the system managed the attachment and to minimize the effect of the message body in the test results.

8.3 Message Sending

To optimize the performance of the test as well as eliminate possible human error that could occur with testing, the test system employed an automated mechanism to send the test messages. This consisted of a Web page written for ASP.NET that allowed the test engineer to send any of the messages on the matrix to any of the test accounts being used for the testing.

The Web page had three elements. The first was a list-select box displaying the three message body sizes from which the test engineer could select. The next element on the page allowed the test engineer to select the attachment type. Finally, the page contained a text box to allow the test engineer to input the e-mail address that would receive the message as well as a submit button to send the test message.

When the test engineer submitted the message, the underlying ASP.NET script—utilizing the Sytem.Net.Mail APIs available via the .NET runtime—created the message by pasting the contents of the selected text file into the message body and then attaching the selected file, if any, to the final message. The message then traveled to the specified recipient. This control page allowed the engineer to easily send any of the given messages on the matrix efficiently and without having to navigate through multiple steps using a mail client such as Microsoft Outlook or Microsoft Outlook Web Access.

8.4 Data Analysis

To analyze the efficiency of each messaging system, the test system captured the amount of data transferred. To do this, we used a Microsoft Windows-based packet capture utility, Ethereal, running on a Windows 2003 server.

Ethereal allowed the test engineer to capture all the data transferred over a single physical network. Once Ethereal captured the data, the test engineer could view it on a packet-by-packet basis, allowing an analysis of exactly how much data the wireless e-mail system transferred to communicate an e-mail message.

To automate the testing process and help eliminate potential human error, we wrote a test application using the Microsoft .NET platform. This test application took a single Ethereal output log as its input and reported the total number of bytes transferred to and from the device. It also reported at which layers this data transfer occurred (e.g., TCP, IP, Ethernet, etc.). Using this utility, the test engineer was able to easily and accurately determine the amount of data transferred in any specific test. For each message type in the test matrix, we did five separate runs and then averaged them to produce the test result summary. As shown in the actual test results in Appendix C of this document, the runs were highly consistent with each other. This allowed us to conclude that five runs was a sufficient number to accurately represent the performance of these systems.

8.5 Microsoft Direct Push Environment

Testing the data throughput with the Windows Mobile device with Direct Push was quite straightforward; all components required for the testing could be hosted on the same physical network and configured in-house. The client device for these tests was a T-Mobile MDA. To work with Direct Push, we had to update the ROM on the device to a version created by i-mate™. This was necessary because at the time we did the testing, a U.S. version of the MDA that supported Direct Push was not yet available. We also configured the Outlook mobile e-mail settings to work with the test Exchange Server, which required a test account on the server that supported mobile access and Direct Push. We configured a PC with Microsoft Outlook to access the mobile test account and then synchronized the mobile device to the PC using ActiveSync.

Finally, using the mobile connection manager on the device, we enabled Direct Push. The connection manager is an applet located in the system properties folder of the device. For actual e-mail reception, we set up the device to communicate with the Agilent 8960 inside a shielded transmitter box. This allowed the device to connect at an RF level with the test system but prevented it from receiving any other RF signals that may have been present in the environment. At this point, the device could communicate via the Agilent 8960 with the Exchange Sever.

Once we established this connection, the test engineer could send messages to the test account and capture traffic data to and from the handheld. The following figure illustrates the test configuration.

Figure 1: Microsoft Direct Push Setup

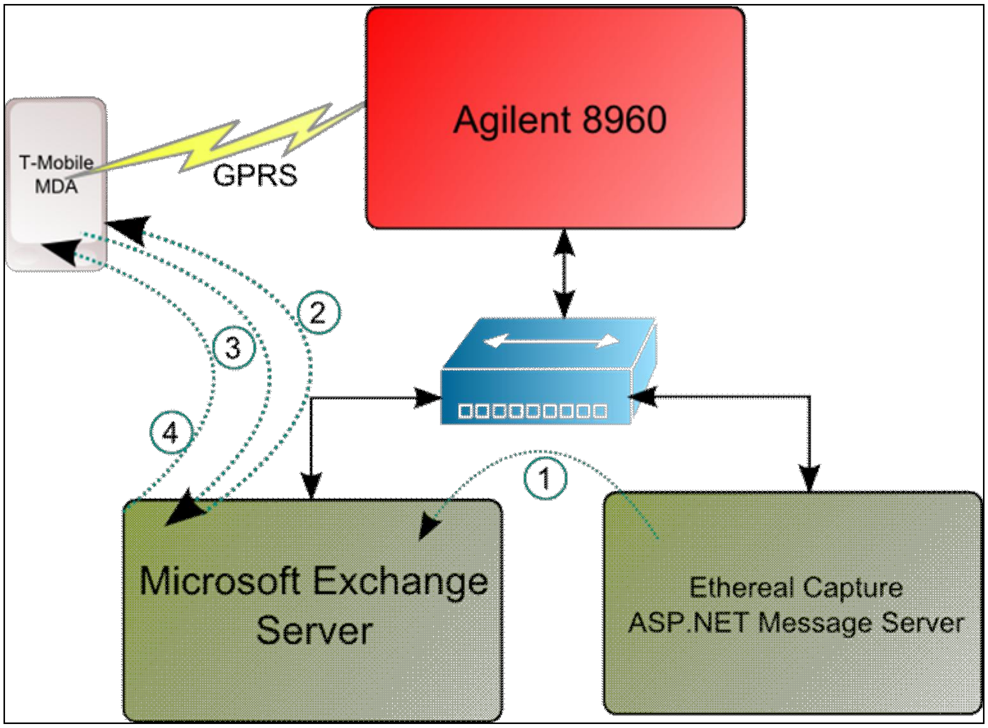


Figure 1 Legend

1. E-mail sent from test script to Exchange Server
2. Exchange Server notifies handheld of new message
3. Handheld requests new message from Exchange Server
4. Exchange Server sends new message to handheld

8.6 RIM BlackBerry Environment

Testing with the BlackBerry configuration required a more complex test configuration because it needed to include the RIM NOC in the communications path. Normally, messages sent to a mobile user flow from the Exchange Server to the BlackBerry Enterprise Server to the RIM NOC to the cellular operator, and finally to the mobile device. Messages from a mobile user flow in the opposite direction. Our test configuration emulated this entire communications path, as shown in the figure below. We used a Virtual Private Network (VPN) between our test network and the RIM NOC that carried both the BES to NOC communications and the NOC to wireless system communications. In commercial networks, this occurs across two separate networks.

Figure 2: RIM BlackBerry Setup

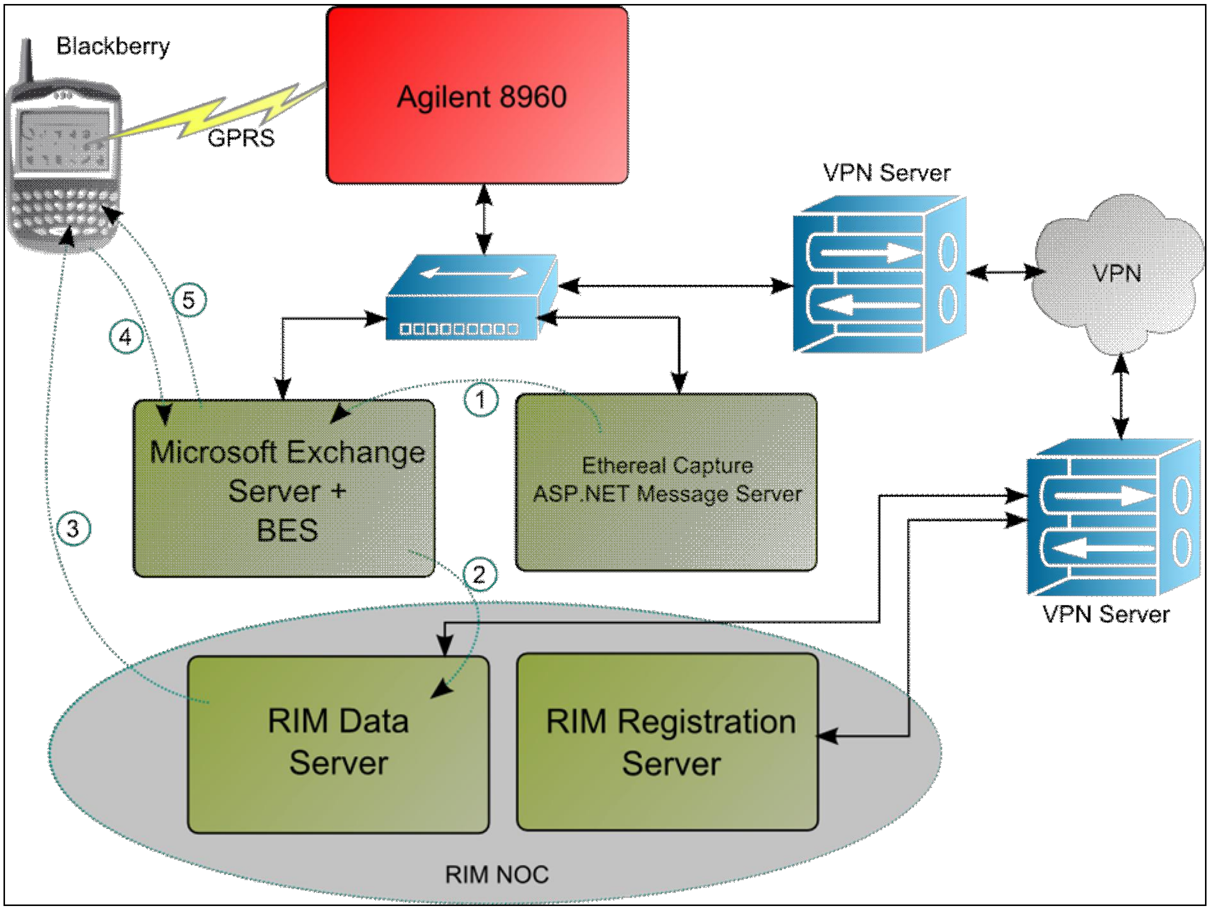


Figure 2 Legend

1. E-mail sent from test script to Exchange Server
2. The BlackBerry Enterprise Server is notified of the new message and notifies the data server at the RIM NOC of the new message
3. The data server at the RIM NOC then notifies the handheld that a new message has arrived
4. The handheld and the RIM NOC communicate to confirm availability
5. The BES sends the new message to the handheld

For the VPN server on the test network, we used a Cisco 850 broadband router configured to support a VPN tunnel with the RIM development VPN concentrator. Once we established the VPN tunnel, we configured the BES to communicate over the VPN tunnel with the RIM NOC.

We then created an e-mail account on the Exchange Server and used the BlackBerry Desktop Manager to configure the handheld to use this account. At this point, the BlackBerry 7290 could communicate over RF via the Agilent 8960 with the RIM NOC and the local BES. As with the Microsoft Direct Push test, we used the shielded transmitter box to block unwanted radio signals. Once we had the connection established, the test engineer could send messages to the test account and capture traffic data to and from the handheld.

9 Appendix C: Detailed Test Results

The following sections show the actual measurements for each test, including all five test passes. Italics show the average values. A figure illustrates the results.

9.1 5 KB Text Message, No Attachment

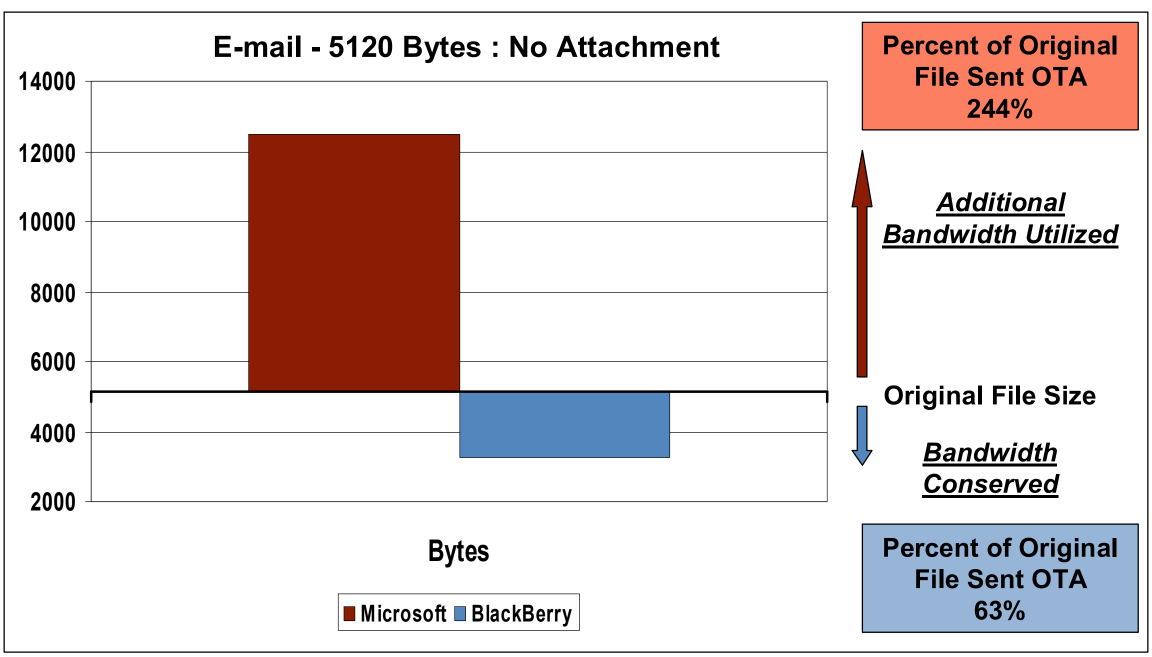
Microsoft

Body size	Attach size	Attach type	Upload - init msg	Download - init msg	Upload - full msg	Download - full msg	Upload - attach	Download - attach	Total upload	Total download	Total bytes
5120	0	n/a	2240	2687	2560	5104	0	0	4800	7791	12591
5120	0	n/a	2282	2747	2560	5104	0	0	4842	7851	12693
5120	0	n/a	2198	2627	2476	4984	0	0	4674	7611	12285
5120	0	n/a	2240	2687	2518	5044	0	0	4758	7731	12489
5120	0	n/a	2240	2687	2476	4984	0	0	4716	7671	12387
5120	*0*		*2240*	*2687*	*2518*	*5044*	*0*	*0*	*4758*	*7731*	*12489*

RIM

Body size	Attach size	Attach type	Upload - init msg	Download - init msg	Upload - full msg	Download - full msg	Upload - attach	Download - attach	Total upload	Total download	Total bytes
5120	0	n/a	60	1026	215	1925	0	0	275	2951	3226
5120	0	n/a	60	1034	215	1925	0	0	275	2959	3234
5120	0	n/a	60	1034	215	1925	0	0	275	2959	3234
5120	0	n/a	60	1034	275	1925	0	0	335	2959	3294
5120	0	n/a	60	1034	215	1925	0	0	275	2959	3234
5120	*0*		*60*	*1032.4*	*227*	*1925*	*0*	*0*	*287*	*2957.4*	*3244*

Figure 3: Microsoft Direct Push versus RIM BlackBerry



9.2 10 KB Text Message, No Attachment

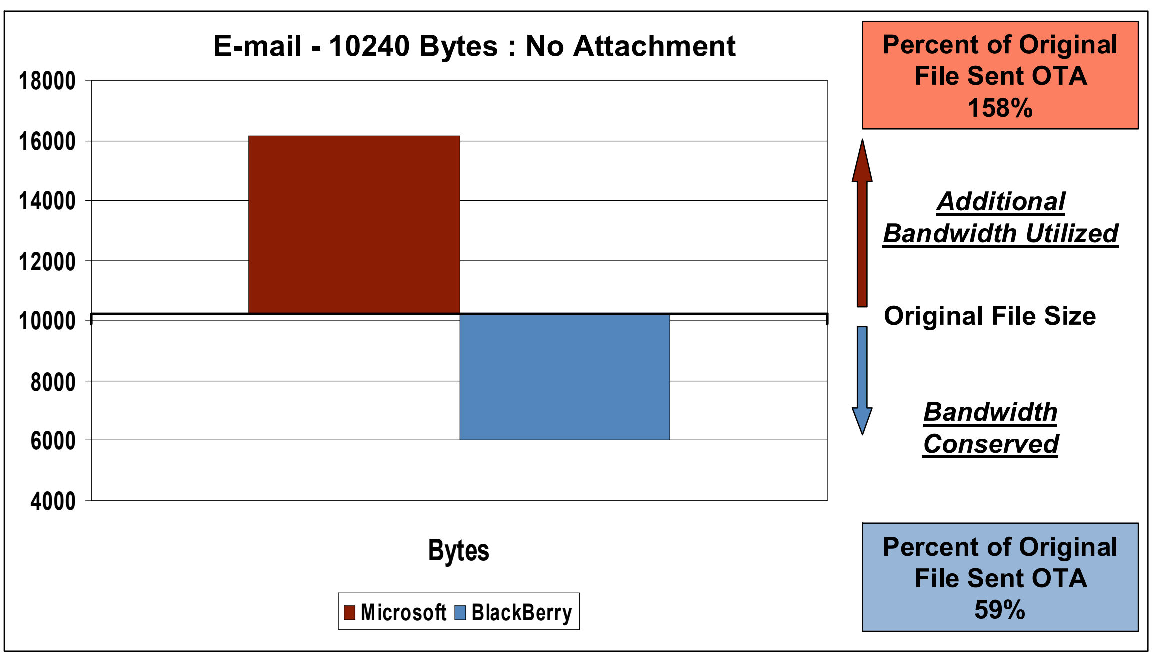
Microsoft

Body size	Attach size	Attach type	Upload - init msg	Download - init msg	Upload - full msg	Download - full msg	Upload - attach	Download - attach	Total upload	Total download	Total bytes
10240	0	n/a	2114	2507	3581	7952	0	0	5695	10459	16154
10240	0	n/a	2114	2507	3581	7952	0	0	5695	10459	16154
10240	0	n/a	2114	2507	3581	7952	0	0	5695	10459	16154
10240	0	n/a	2114	2507	3581	7952	0	0	5695	10459	16154
10240	0	n/a	2114	2507	3581	7952	0	0	5695	10459	16154
10240	*0*		*2114*	*2507*	*3581*	*7952*	*0*	*0*	*5695*	*10459*	*16154*

RIM

Body size	Attach size	Attach type	Upload - init msg	Download - init msg	Upload - full msg	Download - full msg	Upload - attach	Download - attach	Total upload	Total download	Total bytes
10240	0	n/a	60	1026	275	4651	0	0	335	5677	6012
10240	0	n/a	60	1026	275	4651	0	0	335	5677	6012
10240	0	n/a	60	1026	275	4651	0	0	335	5677	6012
10240	0	n/a	60	1026	275	4651	0	0	335	5677	6012
10240	0	n/a	60	1026	275	4651	0	0	335	5677	6012
10240	*0*		*60*	*1026*	*275*	*4651*	*0*	*0*	*335*	*5677*	*6012*

Figure 4: Microsoft Direct Push versus RIM BlackBerry



9.3 20 KB Text Message, No Attachment

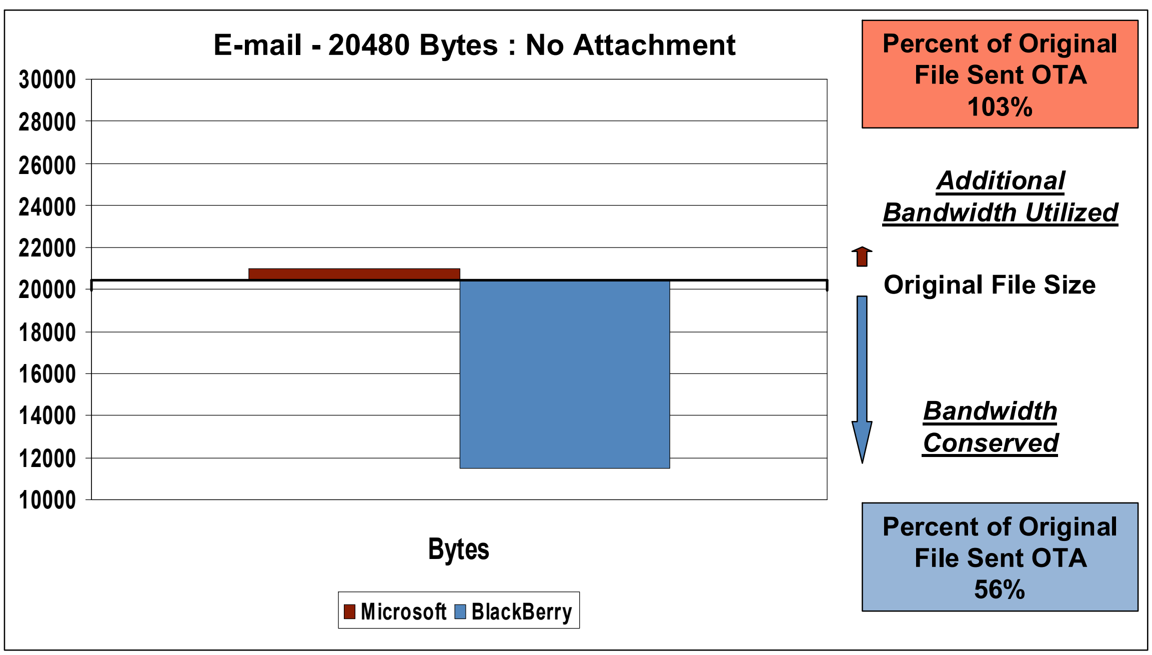
Microsoft

Body size	Attach size	Attach type	Upload - init msg	Download - init msg	Upload - full msg	Download - full msg	Upload - attach	Download - attach	Total upload	Total download	Total bytes
20480	0	n/a	2240	2639	2938	13177	0	0	5178	15816	20994
20480	0	n/a	2240	2639	2938	13177	0	0	5178	15816	20994
20480	0	n/a	2240	2639	2938	13177	0	0	5178	15816	20994
20480	0	n/a	2240	2639	2938	13177	0	0	5178	15816	20994
20480	0	n/a	2240	2639	2938	13177	0	0	5178	15816	20994
20480	*0*		*2240*	*2639*	*2938*	*13177*	*0*	*0*	*5178*	*15816*	*20994*

RIM

Body size	Attach size	Attach type	Upload - init msg	Download - init msg	Upload - full msg	Download - full msg	Upload - attach	Download - attach	Total upload	Total download	Total bytes
20480	0	n/a	60	1010	395	10047	0	0	455	11057	11512
20480	0	n/a	60	1010	395	10055	0	0	455	11065	11520
20480	0	n/a	60	1010	395	10055	0	0	455	11065	11520
20480	0	n/a	60	1010	395	10055	0	0	455	11065	11520
20480	0	n/a	60	1010	395	10055	0	0	455	11065	11520
20480	*0*		*60*	*1010*	*395*	*10053.4*	*0*	*0*	*455*	*11063.4*	*11518.4*

Figure 5: Microsoft Direct Push versus RIM BlackBerry



9.4 5 KB Text Message, 149 KB JPEG Attachment

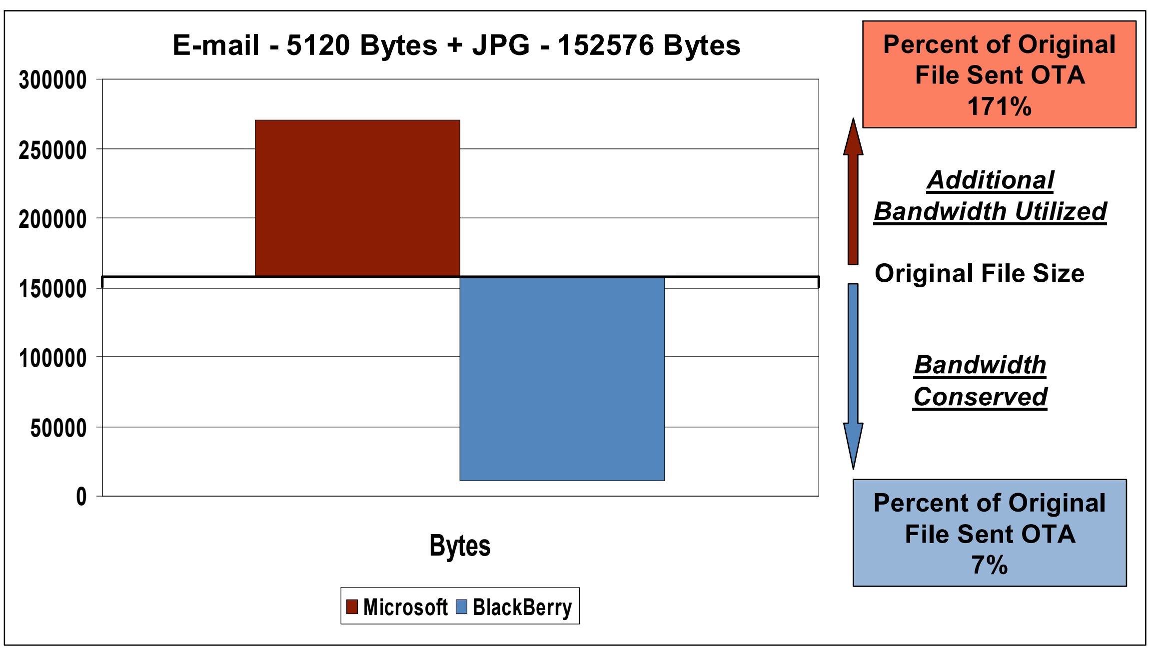
Microsoft

Body size	Attach size	Attach type	Upload - init msg	Download - init msg	Upload - full msg	Download - full msg	Upload - attach	Download - attach	Total upload	Total download	Total bytes
5120	152576	jpeg	2705	2726	2458	5129	15818	241549	20981	249404	270385
5120	152576	jpeg	2705	2726	2458	5129	15818	241549	20981	249404	270385
5120	152576	jpeg	2705	2726	2458	5129	15818	241549	20981	249404	270385
5120	152576	jpeg	2705	2726	2458	5129	15818	241549	20981	249404	270385
5120	152576	jpeg	2705	2726	2458	5129	15818	241549	20981	249404	270385
5120	*152576*		*2705*	*2726*	*2458*	*5129*	*15818*	*241549*	*20981*	*249404*	*270385*

RIM

Body size	Attach size	Attach type	Upload - init msg	Download - init msg	Upload - full msg	Download - full msg	Upload - attach	Download - attach	Total upload	Total download	Total bytes
5120	152576	jpeg	60	1106	215	1885	706	7200	981	10191	11172
5120	152576	jpeg	60	1098	215	1885	706	7200	981	10183	11164
5120	152576	jpeg	60	1106	215	1885	706	7200	981	10191	11172
5120	152576	jpeg	60	1106	215	1885	706	7200	981	10191	11172
5120	152576	jpeg	60	1106	215	1885	706	7200	981	10191	11172
5120	*152576*		*60*	*1104.4*	*215*	*1885*	*706*	*7200*	*981*	*10189.4*	*11170.4*

Figure 6: Microsoft Direct Push versus RIM BlackBerry



9.5 5 KB Text Message, 162 KB PDF Text Attachment

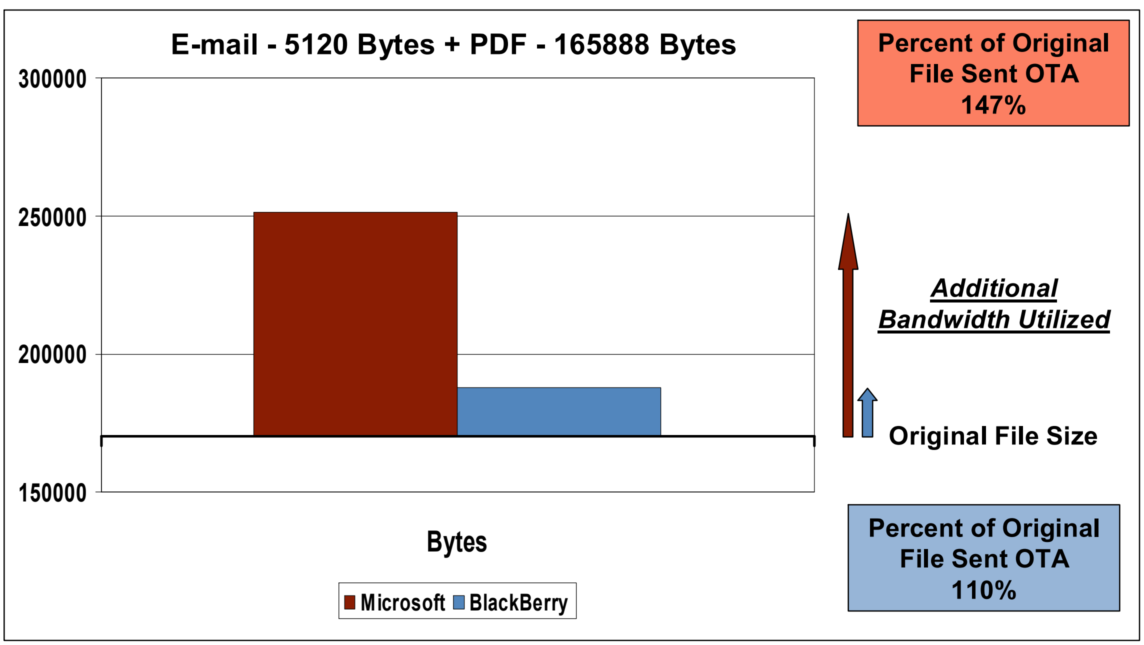
Microsoft

Body size	Attach size	Attach type	Upload - init msg	Download - init msg	Upload - full msg	Download - full msg	Upload - attach	Download - attach	Total upload	Total download	Total bytes
5120	165888	pdf-text	2705	2726	2458	5129	15017	223381	20180	231236	251416
5120	165888	pdf-text	2705	2726	2458	5129	15017	223381	20180	231236	251416
5120	165888	pdf-text	2705	2726	2458	5129	15017	223381	20180	231236	251416
5120	165888	pdf-text	2705	2726	2458	5129	15017	223381	20180	231236	251416
5120	165888	pdf-text	2705	2726	2458	5129	15017	223381	20180	231236	251416
5120	*165888*		*2705*	*2726*	*2458*	*5129*	*15017*	*223381*	*20180*	*231236*	*251416*

RIM

Body size	Attach size	Attach type	Upload - init msg	Download - init msg	Upload - full msg	Download - full msg	Upload - attach	Download - attach	Total upload	Total download	Total bytes
5120	165888	pdf-text	60	1090	275	1885	38920	145360	39255	148335	187590
5120	165888	pdf-text	60	1090	215	1885	38880	145524	39155	148499	187654
5120	165888	pdf-text	60	1090	215	1885	38880	145524	39155	148499	187654
5120	165888	pdf-text	60	1090	215	1885	38880	145524	39155	148499	187654
5120	165888	pdf-text	60	1090	215	1885	38880	145524	39155	148499	187654
5120	*165888*		*60*	*1090*	*227*	*1885*	*38888*	*145491.2*	*39175*	*148466.2*	*187641.2*

Figure 7: Microsoft Direct Push versus RIM BlackBerry



9.6 5 KB Text Message, 106 KB PDF Text and Image Attachment

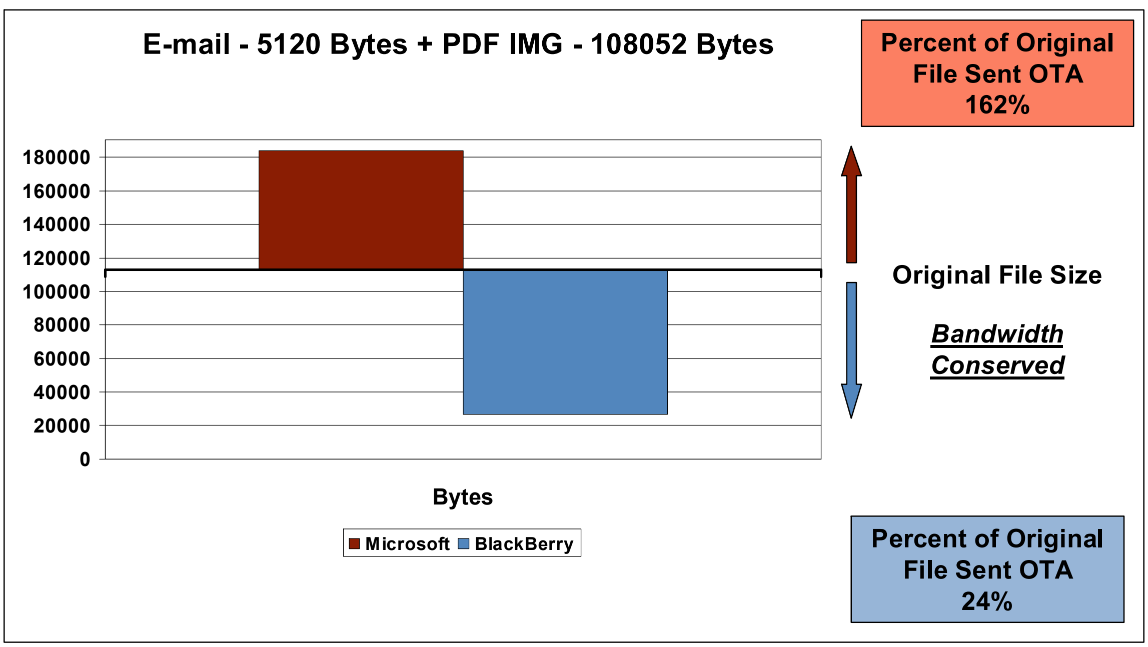
Microsoft

Body size	Attach size	Attach type	Upload - init msg	Download - init msg	Upload - full msg	Download - full msg	Upload - attach	Download - attach	Total upload	Total download	Total bytes
5120	108052	pdf-img	2106	2546	3660	5146	11543	158670	17309	166362	183671
5120	108052	pdf-img	2082	2605	3660	5090	11543	158670	17285	166365	183650
5120	108052	pdf-img	2082	2605	3660	5090	11543	158670	17285	166365	183650
5120	108052	pdf-img	2082	2605	3660	5090	11543	158670	17285	166365	183650
5120	108052	pdf-img	2106	2546	3660	5146	11543	158670	17309	166362	183671
5120	*108052*		*2091.6*	*2581.4*	*3660*	*5112.4*	*11543*	*158670*	*17295*	*166363.8*	*183658.4*

RIM

Body size	Attach size	Attach type	Upload - init msg	Download - init msg	Upload - full msg	Download - full msg	Upload - attach	Download - attach	Total upload	Total download	Total bytes
5120	108052	pdf-img	60	1146	215	2077	2432	20966	2707	24189	26896
5120	108052	pdf-img	60	1146	215	2077	2432	20966	2707	24189	26896
5120	108052	pdf-img	60	1146	215	2077	2432	20966	2707	24189	26896
5120	108052	pdf-img	60	1146	215	2077	2432	20966	2707	24189	26896
5120	108052	pdf-img	60	1146	215	2077	2432	20966	2707	24189	26896
5120	*108052*		*60*	*1146*	*215*	*2077*	*2432*	*20966*	*2707*	*24189*	*26896*

Figure 8: Microsoft Direct Push versus RIM BlackBerry



9.7 5 KB Text Message, 500 KB Word Document Attachment

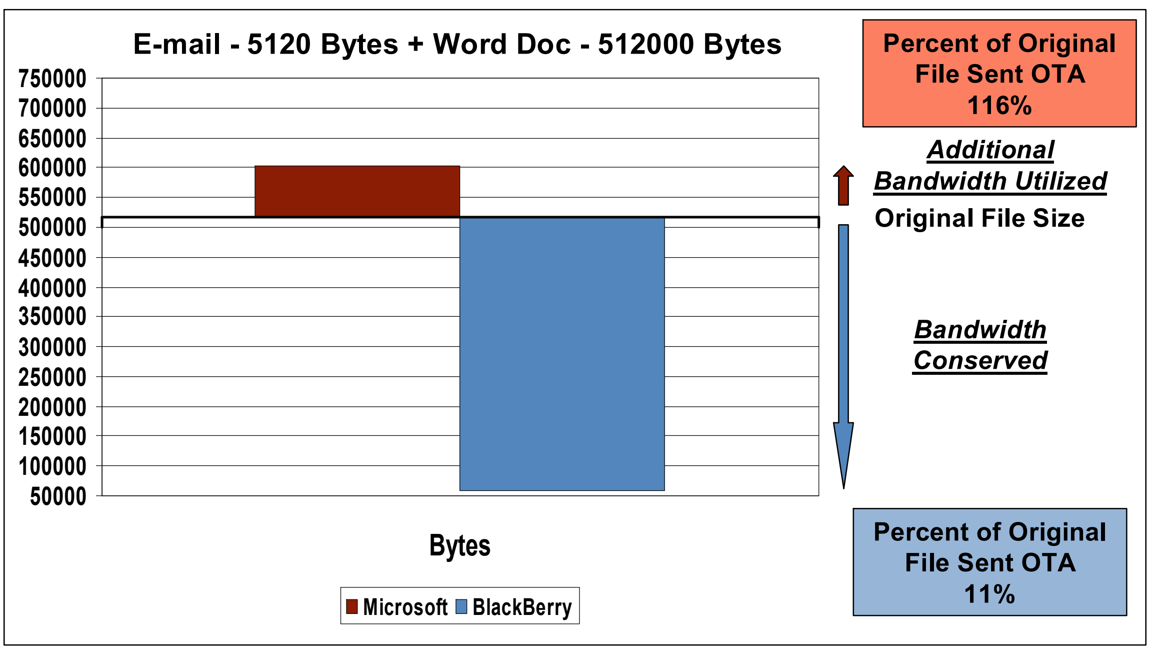
Microsoft

Body size	Attach size	Attach type	Upload - init msg	Download - init msg	Upload - full msg	Download - full msg	Upload - attach	Download - attach	Total upload	Total download	Total bytes
5120	512000	word-doc	2705	2726	2458	5129	33159	555850	38322	563705	602027
5120	512000	word-doc	2705	2726	2458	5129	33159	555850	38322	563705	602027
5120	512000	word-doc	2705	2726	2458	5129	33159	555850	38322	563705	602027
5120	512000	word-doc	2705	2726	2458	5129	33159	555850	38322	563705	602027
5120	512000	word-doc	2705	2726	2458	5129	33159	555850	38322	563705	602027
5120	*512000*		*2705*	*2726*	*2458*	*5129*	*33159*	*555850*	*38322*	*563705*	*602027*

RIM

Body size	Attach size	Attach type	Upload - init msg	Download - init msg	Upload - full msg	Download - full msg	Upload - attach	Download - attach	Total upload	Total download	Total bytes
5120	512000	word-doc	60	1090	215	1885	8189	46757	8464	49732	58196
5120	512000	word-doc	60	1098	215	1885	8189	46788	8464	49771	58235
5120	512000	word-doc	60	1098	215	1885	8189	46757	8464	49740	58204
5120	512000	word-doc	60	1098	215	1885	8189	46757	8464	49740	58204
5120	512000	word-doc	60	1098	215	1885	8189	46757	8464	49740	58204

Figure 9: Microsoft Direct Push versus RIM BlackBerry



9.8 5 KB Text Message, 944 KB PowerPoint Attachment

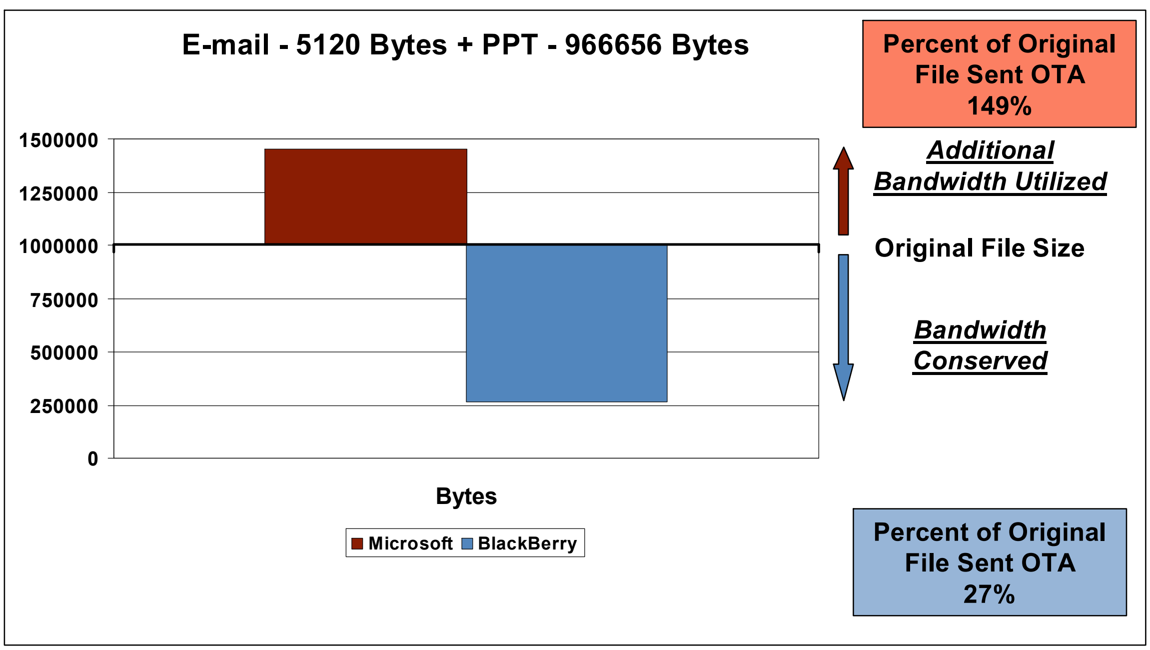
Microsoft

Body size	Attach size	Attach type	Upload - init msg	Download - init msg	Upload - full msg	Download - full msg	Upload - attach	Download - attach	Total upload	Total download	Total bytes
5120	966656	ppt	2705	2726	2458	5129	74663	1362686	79826	1370541	1450367
5120	966656	ppt	2705	2726	2458	5129	74663	1362686	79826	1370541	1450367
5120	966656	ppt	2705	2726	2458	5129	74663	1362686	79826	1370541	1450367
5120	966656	ppt	2705	2726	2458	5129	74663	1362686	79826	1370541	1450367
5120	966656	ppt	2705	2726	2458	5129	74663	1362686	79826	1370541	1450367
5120	*966656*		*2705*	*2726*	*2458*	*5129*	*74663*	*1362686*	*79826*	*1370541*	*1450367*

RIM

Body size	Attach size	Attach type	Upload - init msg	Download - init msg	Upload - full msg	Download - full msg	Upload - attach	Download - attach	Total upload	Total download	Total bytes
5120	966656	ppt	60	1098	215	1877	7512	256187	7787	259162	266949
5120	966656	ppt	60	1098	215	1877	7512	256187	7787	259162	266949
5120	966656	ppt	60	1098	215	1877	7512	256187	7787	259162	266949
5120	966656	ppt	60	1098	215	1877	7512	256187	7787	259162	266949
5120	966656	ppt	60	1098	215	1877	7512	256187	7787	259162	266949
5120	*966656*		*60*	*1098*	*215*	*1877*	*7512*	*256187*	*7787*	*259162*	*266949*

Figure 10: Microsoft Direct Push versus RIM BlackBerry



9.9 5 KB Text Message, 75 KB Excel Attachment

Microsoft

Body size	Attach size	Attach type	Upload - init msg	Download - init msg	Upload - full msg	Download - full msg	Upload - attach	Download - attach	Total upload	Total download	Total bytes
5120	76800	Excel	2174	2657	2452	4505	2788	14417	7414	21579	28993
5120	76800	Excel	2168	2604	2452	4507	2850	14417	7470	21528	28998
5120	76800	Excel	2228	2605	2692	4613	2788	14417	7708	21635	29343
5120	76800	Excel	2174	2658	2512	4507	2788	14417	7474	21582	29056
5120	76800	Excel	2108	2605	2512	4453	2788	14417	7408	21475	28883
5120	*76800*	*Excel*	*2170.4*	*2625.8*	*2524*	*4517*	*2800.4*	*14417*	*7494.8*	*21559.8*	*29054.6*

RIM

Body size	Attach size	Attach type	Upload - init msg	Download - init msg	Upload - full msg	Download - full msg	Upload - attach	Download - attach	Total upload	Total download	Total bytes
5120	76800	Excel	60	1090	215	1885	933	3401	1208	6376	7584
5120	76800	Excel	60	1090	215	1885	933	3401	1208	6376	7584
5120	76800	Excel	60	1090	215	1885	933	3393	1208	6368	7576
5120	76800	Excel	60	1090	215	1885	933	3401	1208	6376	7584
5120	76800	Excel	60	1090	215	1877	933	3401	1208	6368	7576
5120	*76800*		*60*	*1090*	*215*	*1883.4*	*933*	*3399.4*	*1208*	*6372.8*	*7580.8*

Figure 11: Microsoft Direct Push versus RIM BlackBerry