
Fax Overview

Developer's Reference

Fax Overview
Developer's Reference
PN S000265C, Rev. C
11/01/07

Copyright © 2002-7 by Multi-Tech Systems, Inc.

This publication may not be reproduced, in whole or in part, without prior expressed written permission from Multi-Tech Systems, Inc. All rights reserved.

Multi-Tech Systems, Inc. makes no representations or warranties with respect to the contents hereof and specifically disclaims any implied warranties of merchantability or fitness for any particular purpose. Furthermore, Multi-Tech Systems, Inc. reserves the right to revise this publication and to make changes from time to time in the content hereof without obligation of Multi-Tech Systems, Inc. to notify any person or organization of such revisions or changes.

Revisions

Revision Level	Date	Description
A	07/26/02	First release.
B	04/18/07	Updated the Technical Support contact list.
C	11/01/07	Removed France from Technical Support contact list.

Trademarks

Trademarks of Multi-Tech Systems, Inc. are Modem and the Multi-Tech logo.
Other trademarks or trade names mentioned in this publication belong to their respective owners.

World Headquarters

Multi-Tech Systems, Inc.
2205 Wooddale Drive
Mounds View, Minnesota 55112
Phone: 763-785-3500 or 800-328-9717
Fax: 763-785-9874

Technical Support

Country	By Email	By Phone
Europe, Middle East, Africa:	support@multitech.co.uk	+(44) 118 959 7774
U.S., Canada, all others:	support@multitech.com	(800) 972-2439 or (763) 717-5863

Internet Address: <http://www.multitech.com>

Table of Contents

FAX OVERVIEW	4
IMAGE FORMAT	4
IMAGE COMPRESSION	5
<i>End of Lines</i>	5
<i>Example Image</i>	6
<i>MH - Modified Huffman</i>	6
<i>Modified Read (MR)</i>	7
<i>Modified Modified Read (MMR)</i>	7
IMAGE TRANSMISSION	8
<i>Example Fax Session</i>	8
<i>Error Correction Protocol</i>	9
<i>Flow Control</i>	10
HOST COMMUNICATION	11
<i>Other Fax Documents</i>	11
<i>Class 1/1.0</i>	11
<i>Class 1 Example Session – One Page, 14.4K, Non-ECM</i>	12
<i>Class 1.0 Example Session – One Page, 33.6K, ECM</i>	13
<i>Class 2/2.0/2.1</i>	14
<i>Class 2/2.0/2.1 Example Session - One Page, 14.4K, No ECM</i>	15
APPENDIX	16
<i>HDLC / Synchronous Unframed Data</i>	16

Fax Overview

A Fax machine uses a set of protocols to allow the transfer of black and white (now color also) images over regular phone lines. The main protocols describe:

- How the image is to be represented (so that the image encoded on one end can be printed out on the other) .
- How the image data is to be transferred across the phone line.
- How a host computer will talk to the modem.

Table of Protocol Types and Their Associated Standards

Protocol Type	Standard
Image Format	ITU T.4, T.6, T.43, T.85
Image Transmission	ITU T.30
Host Communication	ITU T.31 (Class 1/1.0), T.32 (Class 2.0/2.1)

Image Format

A standard black and white fax image consists of a grid of pixels at fixed resolutions and image sizes compressed in a particular manner. The number of pixels in the image depends on the vertical and horizontal resolutions (pixels/in or pixels/mm) and the size of the image. The size of the image (in bytes) depends on the image dimensions, resolution, content and the form of data compression used. Below is a table describing the different parameters that can be used to describe a fax image. These parameters are negotiated to a lowest common denominator between two fax machines or fax modems.

Image Parameter	Possible Values	Description
Vertical Resolution	Standard - 98 pixels / (inch-25.4mm) Fine - 196 pixels / (inch-25.4mm) 200 pixels/(inch-25.4mm) 300 pixels/(inch-25.4mm) 400 pixels/(inch-25.4mm)	Originally Standard and Fine were the only available vertical resolutions. Since 1992, square resolutions became available, 200x200 , 300x300 , 400x400
Horizontal Resolution	Standard - 204 pixels / (inch-25.4mm) 200 pixels/(inch-25.4mm) 300 pixels/(inch-25.4mm) 400 pixels/(inch-25.4mm)	Originally Standard was the only available horizontal resolution. Since 1992, square resolutions became available, 200x200 , 300x300 , 400x400 .
Horizontal Width	A4 (8.27x11.69 in or 210x297 mm) B4 (10.04x14.33 in or 255x364 mm) A3 (11.93x11.69 in or 303x297 mm) U.S. Letter (8.5x11 in or 215.9x279.4 mm) U.S. Legal (8.5x14 in or 215.9x355.6 mm)	Originally only A4 , B4 , and A3 were the available widths, and this with the old Standard horizontal resolution resulted in page pixel widths of 1728 , 2048 , and 2432 . In 1996, when U.S. Letter and Legal sizes were added are used, only 1728 pixels are sent.
Vertical Height	Unlimited Length A4 (8.27x11.69 in or 210x297 mm) B4 (10.04x14.33 in or 255x364 mm) U.S. Letter (8.5x11 in or 215.9x279.4 mm) U.S. Legal (8.5x14 in or 215.9x355.6 mm)	It used to be that there were only lengths of Unlimited , A4 and B4 lengths. With a Standard vertical resolution, A4 and B4 resulted in 1143 and 1401 pixels respectively. With a Fine vertical resolution this resulted in 2286 and 2802 pixels respectively.

Image Compression

Once the image size and resolution are determined by negotiation with the remote fax machine, the image can be compressed.

In order for faxing to be practical, the raw bitmap image must be compressed. If a standard A4 size page (in fine resolution) were to be sent over the phone line at 9600 baud, it would take 6 minutes and 52 seconds! This is because the image is 1728 pixels wide by 2286 pixels high. Below you can see the calculation of the amount of time it takes to send this image at 9600 baud.

1728 pixels x 2286 pixels = 3,950,208 pixels
 3,950,208 pixels x 1 bit/pixel* = 3,950,208 bits
 3,950,208 bits x 1 sec/9600 bits = 411 seconds
 411 seconds x 1 minute/60 seconds = **6 minutes, 51 seconds!**

* a black and white image only takes one bit for each pixel because a 0 can represent white and a 1 can represent black.

So, you can see that to make digital faxing practical, compression needed to be a component of it.

Table of Compression Options and How They Are Negotiated

Image Parameter	Possible Values	Description
Data Compression	MH - Modified Huffman MR - Modified Read MMR - Modified Modified Read JBig JPEG	Originally MH, MR and MMR were the only standard compressions available (with MMR requiring an error free transfer). Now there is JBig (T.85) for better compression of black and white images and gray scale images and JPEG (T.43) for color images.

Table of Three Standard Compressions Available for Images (MH, MR and MMR)
and What You Can Expect for Corresponding Compression Ratios

Compression Type	Approximate Average Compression Ratio ¹
Modified Huffman	5:1
Modified Read	7.5:1
Modified Modified Read	10:1

To read about the other compression types, see at T.85 and T.43, which describe JBig and JPEG as they relate to fax.

End of Lines

Before discussing the compression algorithms, the issue that impacts these compression schemes must first be covered. Fax does not require that the image data be sent over an error-corrected link. This means that there can be errors in the data received by the receiving fax machine. When there are errors in the data, the image will be decoded in an incorrect fashion from the point of the error until the end. In order to deal with this eventuality, there is a "re-sync" sequence of bits at the end of every line of pixels in the image.

This "re-sync" sequence is called an end-of-line sequence or more accurately a beginning-of-line (BOL) sequence, and it allows the receiving fax machine to recover from a data error in the received image data. It consists of *at least* (this becomes important, as you shall see later) eleven zero bits and a one bit (and an additional bit - called the tag bit - if MR compression is negotiated). There is no encoded sequence of bits that has eleven consecutive zeros in it; this is unique within the image data.

When encoding an image for a particular compression, there is a BOL sequence followed by the compressed line data and then another BOL sequence followed by the next line's compressed data, and so on.

At the end of the fax there is a sequence of six consecutive BOLs. This is called the Return To Control (RTC) sequence. In the case of an MR compressed image, the tag bit on the BOL indicates whether the line following the BOL is an MH compressed line (a one tag bit) or an MR compressed line (a zero tag bit). Below is an example of MR compressed image:

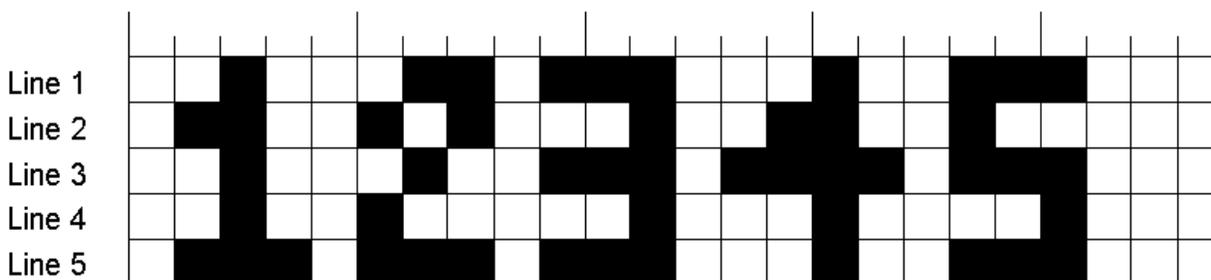
```

BOL(1) - MH compressed line
BOL(0) - MR compressed line
BOL(0) - MR compressed line
BOL(1) - MH compressed line
BOL(0) - MR compressed line
BOL(0) - MR compressed line
...
BOL(1)
BOL(1)
BOL(1)
BOL(1)
BOL(1)
BOL(1)
    
```

When error-correction is used between two fax machines (or modems), then the data will arrive correct or not at all (the data errors are detected and the data is resent, and if the data does not arrive correctly after resending it a number of times, then the modems will hang up), and then the BOL is no longer needed.

Example Image

The following image serves as the example for this section.



MH - Modified Huffman

When data compression schemes were being selected for faxing, an important consideration was the type of images to be transmitted between fax machines. In general, fax machines are used for transmitting business documents which have a fair amount of white space separated by bits of black for the letters encoded on the document. A simple image compression for this type of document would be to describe each line of pixels by listing the runs of white and black. So, in the example image, the runs for the first line would be:

2 white, 1 black, 3 white, 2 black, 1 white, 3 black, 3 white, 1 black, 2 white, 3 black, 3 white.

If you then used the following table to encode the bits, the first line description would be (assuming that the color alternates with each run or set of bits):

10 11 0 10 11 0 0 11 10 0 0

Run Length	Bit Code
1	11
2	10
3	0

This is a total of 17 bits instead of 24 (not a great improvement but you get the idea). Modified Huffman does this except that Modified Huffman research determined that a number of white and black run lengths occurred most often; then they assigned the shortest bit codes to those run lengths. The average compression ratio for a text document is about 5:1. For graphics that have a lot of dithering (where black and white are alternated in an attempt to approximate gray), the compression is perhaps just a little better than 1:1. JBig does a much better job on graphic images (as well as better on text images).

Modified Read (MR)

Modified Huffman is sometimes also referred to as one-dimensional compression because the image is only compressed along one dimension (the horizontal dimension) as each line is compressed independently of each other. Modified Read takes advantage of the fact that in any image (text or graphics) there is generally a high correlation between the content of one line and the next (from line to line the line content probably doesn't change very much). Looking at lines one and two of the example image, there are only 8 columns (out of 24) that are different colors.

The process works by trying to identify which runs of pixels in the *reference line* (the 1st line in this case) corresponds to which lines in the *coding line* (the line being coded or the 2nd line in this case). The run lengths at the beginning of these lines are detailed below:

Line 1: 2 white, 1 black, 3 white, 2 black

Line 2: 1 white, 2 black, 2 white, 1 black

What matters in coding these differences from one line to the next is the starting position of each run. So, if we redo the above run lengths to instead include their starting positions, we can figure out how the runs correspond to each other.

	Run Starting Positions (color)			
Line 1	1(w)	3(b)	4(w)	7(b)
Line 2	1(w)	2(b)	4(w)	6(b)
Difference	0	-1	0	-1
Bit Codes		010	1	010

The first runs always start at one; so, we don't need to encode that. All the others start within one pixel of the run above it. Therefore, all we need to encode is just the -1 and the 0, with the zero bit code being shorter than the -1 code, which, in turn, should be shorter than the -2 code, etc. This is how MR works. If the runs start within 3 pixels to the left or right of the run above it, then only the difference between the starts of the runs is encoded. If it is larger than that, then other rules go into effect to try to "sync" up the runs again so that this kind of coding can take place again.

Because errors in the reference line could invalidate the data in the coding line, there is a maximum number of MR lines that can be encoded before an MH line would be coded again. For a **Standard** resolution fax, the number is one MR for every MH. For a **Fine** or higher resolution fax, the number is two MR for every MH line.

Modified Modified Read (MMR)

MMR is really just a simple extension of MH and MR. This requires (and takes advantage of) that the image be transferred over an error-corrected link so that there are no errors in the image data when received by the receiving fax machine. Since there are no errors in the data, all lines are encoded using the two-dimensional compression described above (for MR). There is no need for BOLs; therefore, they are omitted. The reference line for the first line of the fax is a blank line (since every two dimensional compression of a line needs a reference line).

Image Transmission

Now that we know how to construct the image so that the receiving fax machine can print it out correctly, we need to know how to get the image across the phone line to the receiving fax machine. The standard T.30 describes how this process works. In general, it consists of:

- An exchange of packets of information using a low speed control channel.
- Images sent using a high-speed data channel.

In pre-V.34 fax days (also known as Group 3 fax or G3 fax), the low speed control channel was 300 baud channel 2 and the high speed data channel was from 2400 baud to 14,400 baud using modulations V.27(2400), V.27ter(4800), V.29(7200-9600), and V.17(7200-14400). When connecting in V.34 mode (also known as Super G3 fax), the low speed channel is 1200 or 2400 baud, and the high-speed control channel is anywhere from 2400 to 33600 baud.

Example Fax Session

Below is an example of a fax conversation, which shows the sending of a one-page fax. Real fax time increases with each step down the table. Included in parentheses are the acronyms (defined in T.30) for the type of signal that is sent. Modifications to this procedure for V.34 fax operation are indicated in *Italics*.

Channel	Sender Action	Receiver Action	Description
	Dial Phone #		
	Send Calling Tone (CNG)		1300 Hz tone (on for .5 seconds, off for 3 seconds)
		Answer Phone	
		Send Answer Tone (CED)	2100 Hz tone. <i>If the receiving fax machine is V.34 fax capable, then the 2100 Hz tone is amplitude modulation (its called ANSam). It is here that the control and primary channel data rates are established.</i>
Low Speed		Send Parameter Capabilities and Identifier (DIS/CSI)	This describes what the fax machine can do and what is the ID of the fax machine (usually its phone number or company name).
	75ms Silence		<i>Not used in V.34 mode</i>
Low Speed	Send Negotiated Parameters and Identifier (DCS/TSI)		This is the lowest common denominator between the capabilities of both machines and the ID of the sending machine.
	75ms Silence		<i>Not used in V.34 mode</i>
High Speed	1.5 seconds of test data for High Speed channel (TRN)		This tests to see if the line can send at the high-speed rate that was negotiated without too many errors. <i>This step is omitted for V.34 faxes.</i>
		Test Results (CFR or FTT)	This tells the sender how the test data came out. A CFR is sent if it was OK, and an FTT is sent if not. <i>A CFR is always sent in V.34 mode.</i>
High Speed	The image data		This happens if a CFR is received other wise the DCS/TRN is resent. <i>If error correction is negotiated (it is mandatory for V.34 fax) then the</i>

Channel	Sender Action	Receiver Action	Description
			image data is sent using the error correction protocol (described below).
	75ms Silence		<i>Not used in V.34 mode</i>
Low Speed	Send Next Page Intentions (EOP, MPS, EOM)		This tells the receiving fax machine what the sending fax machine intends to do after this page. An EOP is sent if this is the last page. An MPS is sent if it is not, and an EOM is sent if the sender wants to re-negotiate the session parameters (and send another page besides).
Low Speed		Page Results (MCF, RTP, RTN)	This tells the sender if the page received was of good quality or not. This can be calculated by seeing how many lines were received that were not the negotiated width. An MCF is sent if the page is of good quality, a RTP is sent if a page was of OK quality, and an RTN is sent if the page is of bad quality.
Low Speed	Send Disconnect (DCN)		Assuming the page was received with good quality, then the sender sends a disconnect packet.

Acronym Key:

CNG - CalliNG tone

CED - Called tone

DIS - Digital Information Signal

CSI - Called Subscriber Identifier

DCS - Digital Command Signal

TSI - Transmitting Subscriber Identifier

TRN - TRaiNing signal

CFR - ConFirmation to Receive

FTT - Failure To Train

EOP - End Of Procedure

EOM - End Of Message

MPS - Multi-Page Signal

MCF - Message ConFirmation

RTP - ReTrain Positive

RTN - ReTrain Negative

Error Correction Protocol

Because fax is half-duplex, the most efficient manner of correcting the erred image information is to send a bunch (called a partial page) of data and then have the receiver indicate what parts of it were in error. Then the sender can re-send them. The size of that bunch of data is either 16K or 64K. Both fax machines must be able to do 64K for that to be negotiated. When 16K is used, the data is split into 64-byte frames (each completes its own CRC) and 64K is used. Then 256 bytes frames are used. After the partial page is sent, the receiver sends back a low speed frame that indicates which frames were in error. The sender then resends just those frames. The receiver again checks the frames, and the process continues until either all the data is corrected or the sender decides to give up.

Flow Control

Since there can be limits on how fast a fax machine can receive (and print or store) the image information that is sent from the sending fax machine, there are methods of slowing down the data in order for the receiver to keep up.

When error correction is not used, the receiver has no way to tell the sender to slow down when it is sending the image data. The sender is simply sending and not listening. So, the method to slow the data flow in this case is to allow the receiver to negotiate a minimum amount of time that the transmission of one line of pixels from the image should take. If the receiving fax machine takes 10 milliseconds to print one line of the image, then it could require the sender to send extra useless information (called fill) at the end of each line of the image. This fill is composed of zero bytes inserted at the end of the line, which look like extra zero bits before the next BOL (beginning of line), which is defined as at least eleven zero bits followed by a one bit (and a tag bit if MR coding is used).

When error correction is used, the receiver can tell the sender in between partial pages that it is not ready for more data, and the sender will keep querying until the receiver says it is ready. Because the receiver can only flow-control between either 16K or 64K blocks of data, it must have the ability to save the entire block in memory in real-time so that no data is lost. When the receiver is ready to receive another block, it will notify the sender, and then the sender will send the next partial page. Because this mechanism exists, fill is no longer necessary in the image.

Host Communication

Now that we know how to format the image and how it is really transmitted over the phone lines, we can tackle how we talk a modem into sending or receiving a fax. There are primarily two methods of doing this:

1. Having the host computer implement the T.30 protocol
2. Having the modem implement the T.30 protocol

The first choice is implemented by using the Class 1/1.0 set of commands in the modem. In that case, the modem is used primarily as a modulator/demodulator (AKA modem) of digital data to/from analog sounds over the phone line with very little other processing done in the modem. The host computer tells the modem what modulation to use and then it sends the data to it or receives data from it. Class 1/1.0 consists of just a few commands and is very easy to learn. But writing the code to control the modem is very complex because now the host has to implement all of T.30 (which despite my simplification above is really quite involved) and also test this implementation against many real world fax machines. That said, Class 1/1.0 allows the host to add any features that T.30 supports without relying on the modem to support a particular feature of T.30. So for example, an application could add error corrected faxes to its list of features by adding support for it in its T.30 implementation, whereas if it relied on Class 2/2.0/2.1 modems it they might be out of luck, because most don't support error correction.

The second choice is implemented using Class 2/2.0/2.1 set of commands. In this case, the commands to learn to send a fax through the modem are more complex than Class 1/1.0, but you can leave the burden of compliance with the T.30 specification with the modem. Because the modem is implementing T.30, the host does not have to be as concerned with the timing aspects of T.30 the way that a host implementing T.30 does. For this reason, most fax server software solutions do not like to use Class 1/1.0 modems because they are running on a multi-tasking operating system (such as Unix or NT) that can't guarantee a process not be swapped out for a period of time.

Other Fax Documents

For more information on Class 1/1.0 and Class 2/2.0/2.1 commands, see the Class 1/1.0 and Class 2/2.0/2.1 Developer's Guides also posted to the Multi-Tech Web site.

Class 1/1.0

Class 1/1.0 commands:

Command	Description
AT+FTS	Transmit silence for a period of time
AT+FRS	Listen for silence
AT+FTH	Transmit HDLC framed data at the low speed control channel rate
AT+FRH	Receive HDLC framed data at the low speed control channel rate
AT+FTM	Transmit synchronous unframed image data at given high speed primary channel rate
AT+FRM	Receive synchronous unframed image data at given high speed primary channel rate

For more information on the differences between HDLC framed data and synchronous unframed data, see the appendix. For now, it is enough to know is that HDLC is required for low speed information and for implementing the error-correction image protocol. The synchronous unframed data is used to send/receive non-error corrected image data.

Class 1 Example Session – One Page, 14.4K, Non-ECM

Sending Host	Sending Modem	Description	Receiving Host	Receiving Modem
AT+FCLASS=1				AT+FCLASS=1
	OK		OK	
ATDTnnnn				
			RING	
	CONNECT		CONNECT	
	<CSI frame>	<-- CSI		<CSI frame>
	OK			
AT+FRH=3				
	CONNECT		CONNECT	
	<DIS frame>	<-- DIS		<DIS frame>
	OK		OK	
AT+FTH=3				AT+FRH=3
	CONNECT		CONNECT	
<TSI frame>		TSI -->	<TSI frame>	
			OK	
				AT+FRH=3
	CONNECT		CONNECT	
<DCS frame>		DCS -->	<DCS frame>	
	OK		OK	
AT+FTM=145				AT+FRM=145
	CONNECT		CONNECT	
<TRN data>		TRN -->	<TRN data>	
	OK		OK	
AT+FRH=3				AT+FTH=3
	CONNECT		CONNECT	
	<CFR frame>	<-- CFR		<CFR frame>
	OK		OK	
AT+FTM=146				AT+FRM=146
	CONNECT		CONNECT	
<image data>		image -->	<image data>	
	OK		OK	
AT+FTH=3				AT+FRH=3
	CONNECT		CONNECT	
<EOP frame>		EOP -->	<EOP frame>	
	OK		OK	
AT+FRH=3				AT+FTH=3
	CONNECT		CONNECT	
	<MCF frame>	<-- MCF		<MCF frame>
	OK		OK	
AT+FTH=3				AT+FRH=3
	CONNECT		CONNECT	
<DCN frame>		DCN -->	<DCN frame>	

Class 1.0 Example Session – One Page, 33.6K, ECM

Sending Host	Sending Modem	Description	Receiving Host	Receiving Modem
AT+F34=14,4,1				AT+F34=14,4,1
	OK		OK	
AT+FCLASS=1.0				AT+FCLASS=1.0
	OK		OK	
ATDTnnnn				
			RING	
	+F34:14,1		+F34:14,1	
	CONNECT		CONNECT	
	<DLE><CTRL> <DLE><33.6> <DLE><1.2>	Report to DTE which channel is active and high speed and low speed channel rates.	<DLE><CTRL> <DLE><33.6> <DLE><1.2>	
	<CSI frame>	<-- CSI		<CSI frame>
	<DIS frame>	<-- DIS		<DIS frame>
<TSI frame>		TSI -->	<TSI frame>	
<DCS frame>		DCS -->	<DCS frame>	
	<CFR frame>	<-- CFR		<CFR frame>
<DLE><PRI>		switch to primary channel		
	<DLE><PRI> <DLE><33.6>		<DLE><PRI> <DLE><33.6>	
<image data>		image -->	<image data>	
<DLE><CTRL>		switch to control channel		
	<DLE><CTRL> <DLE><33.6> <DLE><1.2>	Report to DTE which channel is active and high speed and low speed channel rates.	<DLE><CTRL> <DLE><33.6> <DLE><1.2>	
<EOP frame>		EOP -->	<EOP frame>	
	<MCF frame>	<-- MCF		<MCF frame>
<DCN frame>		DCN -->	<DCN frame>	
	OK		OK	

Class 2/2.0/2.1

Class 2/2.0/2.1 implements many commands. The commands are used to configure how the modem will implement the T.30 protocol. Class 2 and Class 2.0/2.1 are very similar except that almost all the command/response names are different (the reasons for this were largely political but that is how it now is). So for most any command/response in Class 2, there is a corresponding command/response in Class 2.0/2.1 with a shorter name. There are some commands in Class 2.0/2.1 that do the work of several commands in Class 2 and the way the end of the page procedure is done is different in Class 2 and 2.0/2.1. The other major difference is that Class 2 is frozen in time and Class 2.0/2.1 has been updated as T.30 has been updated.

The difference between Class 2.0 and 2.1 is that 2.1 supports V.34 speeds. It is actually the same command set, but if the software sets the modem to +FCLASS=2.0 and a call is made to a V.34 fax (also known as Super G3) capable fax machine, only a non V.34 fax call is negotiated. Setting the modem to class 2.0 tells the modem that the software would not understand the added responses that would accompany a V.34 fax call.

Class 2/2.0/2.1 Example Session - One Page, 14.4K, No ECM

In this example all differences when sending a fax in 2.0 mode are indicated in *Italics*.

Sending Host	Sending Modem	Description	Receiving Host	Receiving Modem
AT+FCLASS=2 <i>(2.0)</i>			AT+FCLASS=2 <i>(2.0)</i>	
	OK		OK	
ATDTnnnn			RING	
	+FCO (<i>+FCO</i>)			
	+FCSI:"Receiver " <i>(+FCI)</i>	<-- CSI		
	+FDIS:1,0,5,0,0,0,0 <i>(+FIS)</i>	<-- DIS		
	OK			
AT+FDT			+FCO (<i>+FCO</i>)	
		TSI -->	+FTSI:"Sender " <i>(+FTI)</i>	
		DCS -->	+FDCS:1,0,5,0,0,0, 0,0 (<i>+FCS</i>)	
		TRN -->	OK	
				AT+FDR
	+FDCS:1,0,5,0,0,0,0 <i>(+FCS)</i>	<-- CFR		
	CONNECT		CONNECT	
<image data> <i>(add</i> <DLE><EOP>)		image -->	<image data>	
	OK <i>(not done for</i> <i>2.0/2.1)</i>		+FPTS:1,2155,0,0, 0 (<i>+FPS</i>)	
AT+FET=2 <i>(not</i> <i>done for 2.0/2.1)</i>				
		EOP -->	+FET:2	
			OK	
				AT+FDR
	+FPTS:1 <i>(Not done in</i> <i>Class 2.0/2.1. Bad</i> <i>pages are indicated</i> <i>below by an ERROR</i> <i>response)</i>	<-- MCF		
	+FHNG:0 (<i>+FHS</i>)	DCN -->	+FHNG:0 (<i>+FHS</i>)	
	OK		OK	

Appendix

HDLC / Synchronous Unframed Data

Both HDLC and Synchronous Unframed data are methods of transmitting/receiving synchronous data. Synchronous data means that for each data bit transmitted, a clock is transmitted with it so the receiver knows when to sample the receive-data line. The problem with synchronous data is that you don't know where the byte boundaries are in the data. To solve this you need a way to synchronize the receiver with the bit stream so that it builds bytes up correctly.

HDLC solves this problem by transmitting a particular pattern of bits (known as Flags -- bit sequence 01111110) which signify that there is no "real" data is being transmitted. The receiver can use this to determine when there is an idle line. The transmitter must then make sure that this sequence of bits never occurs in actual data. It does this by inserting a zero after every 5 consecutive one bits, when it is transmitting actual data. The receiver performs the opposite process when it is receiving actual data. In HDLC data is transmitted in frames of data (not sent one byte at a time with a variable amount of idle time between bytes as in asynchronous mode) to reduce the overhead of a full 8 bits to signify an idle line. So typically a conversation is a period of flags followed by a frame of several or many bytes, followed by a CRC, and then followed by flags again as the cycle restarts.

In synchronous unframed data, all the bits transmitted are actual data. In the case of a non-error corrected fax image, the beginning of line sequence (BOL) serves as a synchronizer for the receiver. All data received up to the first BOL is ignored. After the first BOL is received, the receiver is then in sync with the transmitter.

HDLC is used for all control information and for image information when a fax is in error corrected mode. Synchronous unframed mode is used for non-error corrected fax image information.