

## Poor Man's Attachments

Jim Oberhofer KN6PE

October 21, 2009

### The Background

One of the nice features with commercial email is the ability to attach one or more files to the message to be recovered at the receiving end. While this is now second nature for email users, doing this with packet, at 1200 baud no less, is something to be approached carefully.

A future release of Outpost may include native support for attachments and comply with the B2F forwarding protocol now used by Winlink, F6FBB, and other BBSs. Until then, here's an approach for passing binary files with only some additional key-clicks using WinZip.

### Definitions

**File compression:** the process of squeezing a file down to its smallest form. Most files have redundant patterns of ones and zeros. As stated in the reference below, "File-compression programs simply get rid of the redundancy. Instead of listing a piece of information over and over again, a file-compression algorithm lists that information once and then refers back to it whenever it appears in the original file."

Ref: [www.howstuffworks.com/file-compression.htm](http://www.howstuffworks.com/file-compression.htm)

**Uencode/uudecode:** the process of converting a binary file to a text file (uencode) and a converted text file back into its binary file (uudecode). This file format is needed for Outpost because of its reliance on an ASCII data stream when passing an AX.25 packet message. However, this process comes with a size penalty: 3 bytes of any data (ASCII or binary) is converted into 4 bytes of ASCII. With a very efficient compression algorithm, this cost becomes manageable.

### Restrictions

Outpost currently has a 10,000 character (10Kb) message limit. While this limitation will be lifted to align with the Winlink limitation, it must be taken into account when using this process. So, while the source file may be larger than 10Kb, the compressed and encoded file must be less than this limit.

### Creating Binary Attachments

Here's the process for using WinZip to pass binary files with Outpost.

### Identify the file to send

I used a spreadsheet that could be part of a shelter request for supplies.

With Excel, as soon as you start a new spreadsheet, it automatically costs you 13.5 Kb before you type a thing.

	A	B	C	D	E	F	G	H
1	Shelter:	Quinlan Center						
2	City:	Cupertino, CA						
3	Request:	Material replenishment						
4	Date:	3/24/2007						
5								
6	Item#	Desc	On-Hand	Units	Qty Needed	Need Date	Need Time	
7	1	cots	30	ea	25	18-Mar	18:00	750
8	2	blankets	45	ea	15	18-Mar	18:00	675
9	3	water, 12 oz bottles	200	bottles	100	19-Mar	12:00	20000
10	4	First Aid kits	3	kits	12	ASAP		36
11	5	Toilet paper	50	rolls	250	19-Mar	12:00	12500
12	6	tooth brushes	10	ea	50	ASAP		500
13	7	tooth paste	5	ea	55	ASAP		275
14	8	note pads	0	ea	25	19-Mar	18:00	0
15	9	pencils	0	ea	25	19-Mar	18:00	0
16	10	MRE	10	cases	300	18-Mar	18:00	3000

The spreadsheet

For this file as you see it here, the file size came in at 15 Kb.

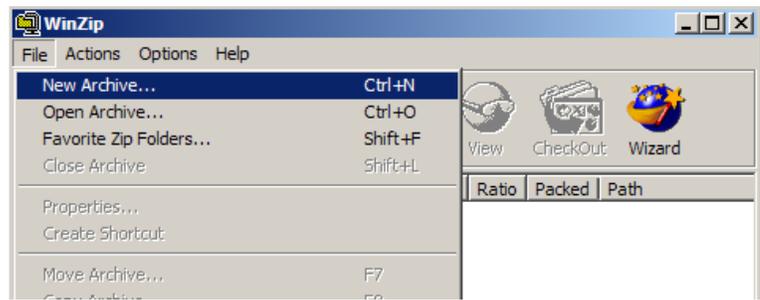


Looking at the Directory

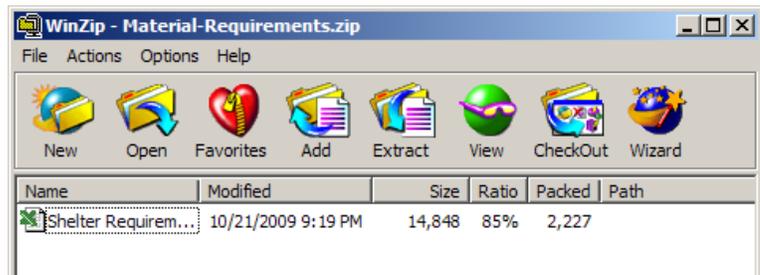
**Zip the file**

1. Run WinZip
2. Create a new archive  
**File > New Archive**

Or, right-click on the file to be compressed and select the **“Add to <filename.zip>** option. Open the newly created .zip file, then continue at Sep 4.



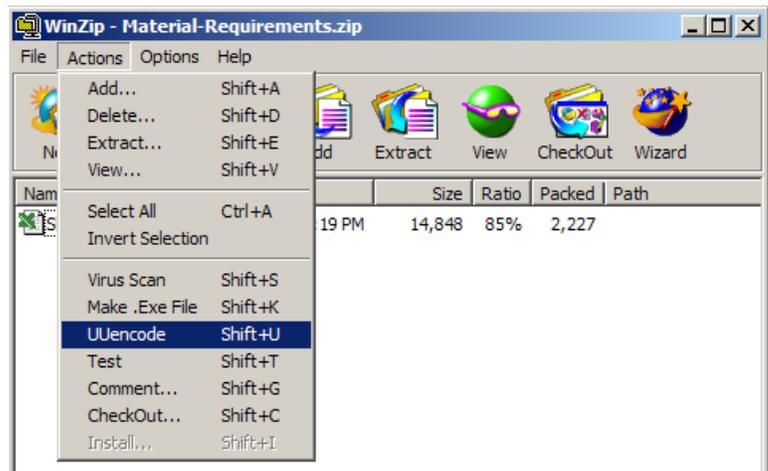
3. Add the file to the Archive. Note the compression. The file size went from 14,868 bytes to 2,227 bytes, an 85% size reduction.



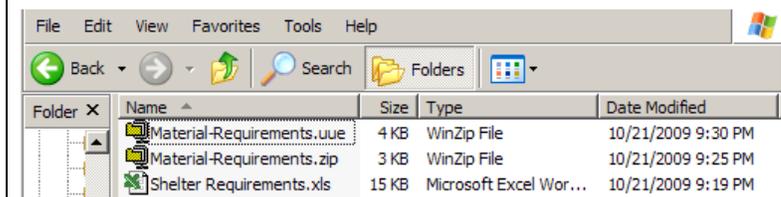
4. Finally, UUencode the file  
**Actions > UUencode**

You will get a message stating that the file was encoded as **Material-Requirements.uue**

5. Exit when done.



Note the size of the .uue file in the directory holding the file.

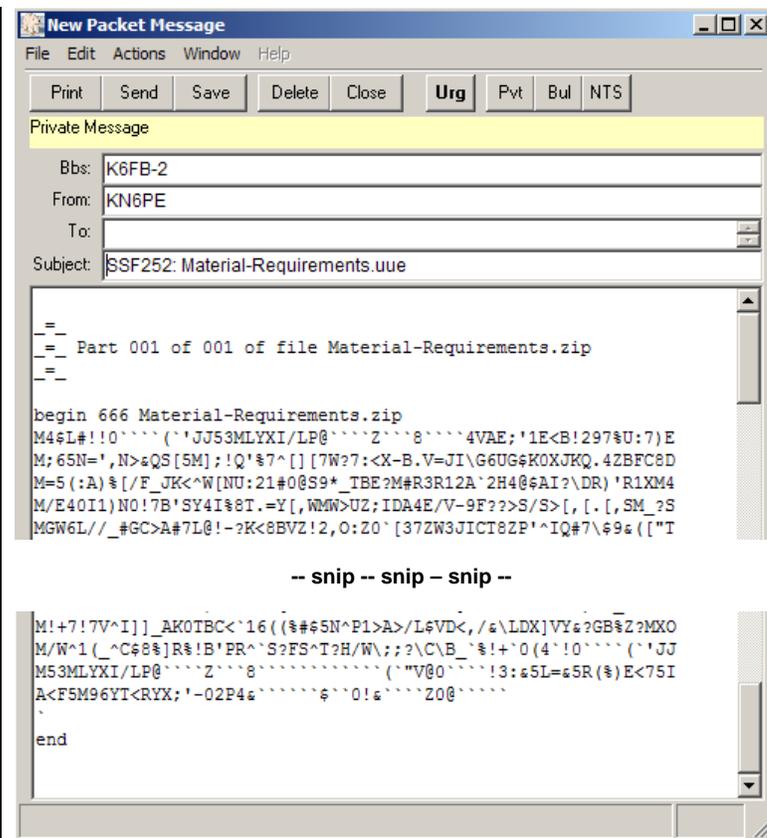


### Creating a Message in Outpost

1. Go to Outpost, and press **New** to open a message form.
2. From the message form, **File > Open** a file.
3. Navigate to the directory holding **Material-Requirements.uue**, and choose the file. Press **OK**
4. The **.uue** file will load and the message will look like this...
5. Fill in the rest of the message as usual.
6. Press **Send**, then **Send/Receive**.

**NOTE:** The file name is added to the subject line. **DO NOT CHANGE THIS!** This will be important on the receiving end.

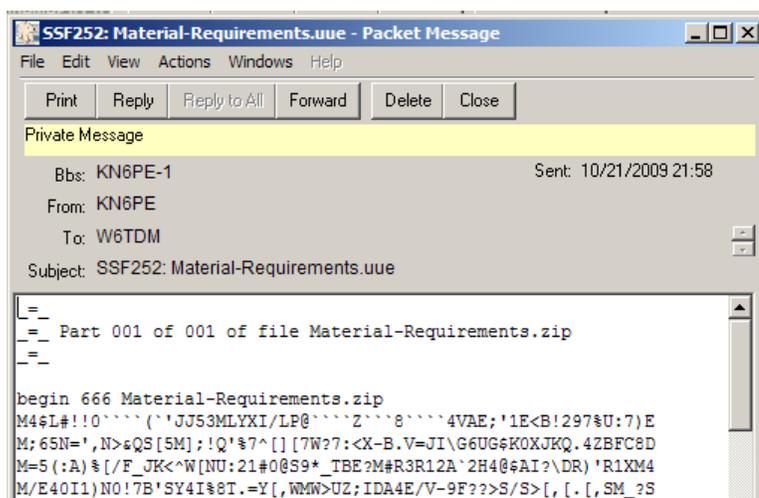
**NOTE:** DO NOT ADD OR DELETE ANYTHING between the top of the 6 header lines to the “end” statement at the bottom.



### Receiving this Message in Outpost

1. At the receiving station, press **Send/Receive** to retrieve your messages.
2. Find this message in the In Tray and open it.

3. Click on **File > Save As...** Outpost defaults the name of the file to the name of the Subject... exactly what we want (.uue included). Put the file where ever you want it. Press **OK**



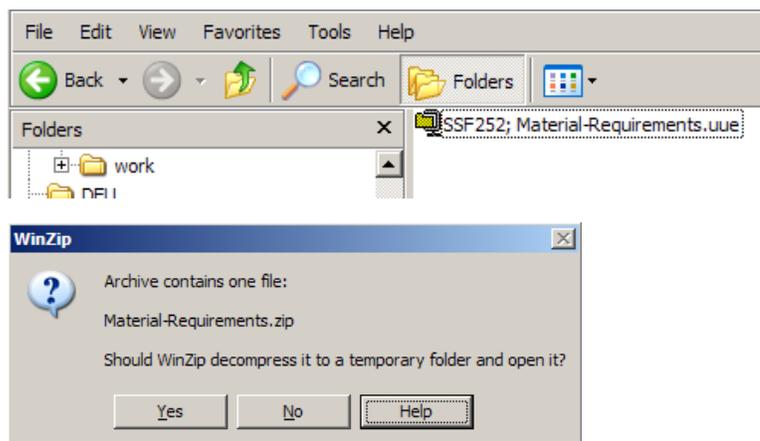
4. Navigate to that directory, and you will see a zip file with the name of the message Subject. Note the WinZip icon.

5. Double-click on the file name. WinZip will ask: "Should WinZip decompress it to a temporary file and open it?"

Click **Yes**.

The file will open and display the spreadsheet.

From Excel, save the spreadsheet file to your PC, or cut and paste the contents wherever you need them.



### Considerations

1. Obviously, this is a very manual process for passing a binary message. You need to watch the size of the message once it is uuencoded to ensure you are within the 10,000 Outpost limit.
2. Passing any large file at 1200 baud will suck the bandwidth from the environment. You should pick your large file transfer opportunities carefully.
3. Not all binary files compress as nicely as the example above. With no special formatting, spreadsheets typically compress down 85%. Word docs much less. Gifs and other picture files even less.
4. You need to check the size after the uuencode step to see if Outpost can support the file size. If it is larger than 10,000, Outpost will complain and truncate the file, thereby making it corrupt at the receiving end.
5. I suppose if you were really desperate to send a file, you could manually cut up the uuencoded file outside of Outpost into something less than 10K blocks, send them all individually, and reassemble them at the receiving end. But this is not for the uninitiated and subject to manual error.
6. If sending single page unformatted spreadsheets, you may be better off saving the spreadsheet as a .csv file (comma-delimited file, also ASCII), and sending that. See [Appnote-0401-Spreadsheet-Data.pdf](#), Method 2 for details.