The SAS Dummy
A SAS® blog for the rest of us

Reading and updating ZIP files with FILENAME ZIP

Chris Hemedinger | JANUARY 29, 2014

In a previous post, I shared an example of using ODS PACKAGE to create ZIP files. But what if you need to *read* a ZIP file within your SAS program? In SAS 9.4, you can use the FILENAME ZIP access method to do the job.

In this example, let's pretend that I need to analyze data that a government agency published (maybe by using SAS!) into a ZIP file. I've selected an *exciting* data source (found via data.gov) about Large Truck Crash Causation.

First, I need to download the latest version of the data file. I'll use PROC HTTP to do that job:

```
/* detect proper delim for UNIX vs. Windows */
%let delim=%sysfunc(ifc(%eval(&sysscp. = WIN),\,/));

/* create a name for our downloaded ZIP */
%let ziploc = %sysfunc(getoption(work))&delim.datafile.zip;
filename download "&ziploc";

/* Download the ZIP file from the Internet*/
proc http
 method='GET'
 url="http://ai.fmcsa.dot.gov/ltccs/Data/TEXT/Public/LTCCS_db_txt_public_01.zip"
 out=download;
run;
```

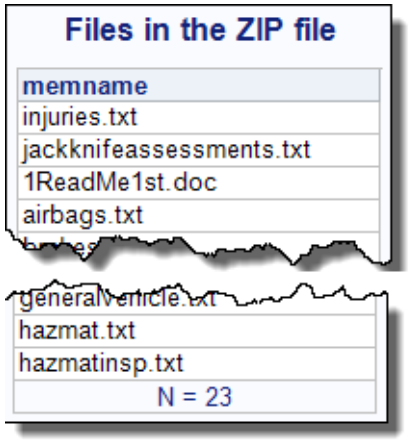
Next, I need to discover what files are within the ZIP file. I'll assign a fileref using the new FILENAME ZIP method. FILENAME ZIP is a directory-based access method, similar to the CATALOG access method or to using FILENAME to map to a folder. You can use functions such as DOPEN and DREAD to treat the ZIP file as if it's a file directory (since that's what it is, in concept).

```
/* Assign a fileref wth the ZIP method */
filename inzip zip "&ziploc";

/* Read the "members" (files) from the ZIP file */
data contents(keep=memname);
 length memname $200;
 fid=dopen("inzip");
 if fid=0 then
  stop;
 memcount=dnum(fid);
 do i=1 to memcount;
  memname=dread(fid,i);
  output;
 end;
 rc=dclose(fid);
run;

/* create a report of the ZIP contents */
title "Files in the ZIP file";
proc print data=contents noobs N;
run;
```

Here's the report of files within the ZIP archive:



Files in the ZIP file

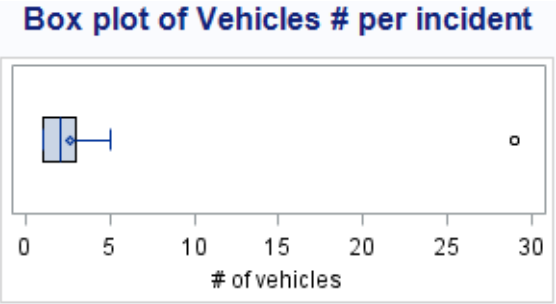
memname
injuries.txt
jackknifeassessments.txt
1ReadMe1st.doc
airbags.txt
generalVehicle.txt
hazmat.txt
hazmatinsp.txt
N = 23

I've identified the HAZMAT.TXT file as the one that I want to analyze. I peeked at the first couple of records and was able to scratch out a simple DATA step to read the data. Notice how I don't need to explicitly extract the HAZMAT.TXT file -- I can simply reference it as a "member" of the INZIP fileref. The ZIP access method does the rest.

```
/* Import a text file directly from the ZIP */
data hazmat;
 infile inzip(hazmat.txt)
   firstobs=2 dsd dlm='09'x;
 input
  CaseID $10.
  VehicleNumber
  Material
  Reportable
  Waiver
  PSU
  PSUStrata
  RATWeight;
run;

title "Box plot of Vehicles # per incident";
ods graphics / height=200 width=450;
proc sgplot data=hazmat;
        hbox vehiclenumber;
        label VehicleNumber="# of vehicles";
        xaxis labelattrs=(size=12) valueattrs=(size=12);
run;
```

SAS reads my data file successfully, and yields this interesting box plot from the SGPLOT step:



(It looks like most "hazardous materials" accidents involved just 2 or 3 vehicles, except for one messy outlier that had nearly 30. Imagine the cleanup effort on that one!)

As an alternative, **if** I know exactly which file I need, I can assign a direct fileref by using the MEMBER= syntax:

```
filename inzip zip "&ziploc" member="hazmat.txt";

/* then my INFILE references the file directly, no parenthesized-member */
data hazmat;
 infile inzip
   firstobs=2 dsd dlm='09'x;
/* ...  */
```

The ZIP access method isn't just for reading. I can also use it to **create and update** ZIP files. For creating ZIP files, I prefer to use ODS PACKAGE. But it's very handy to be able to update ZIP files from a SAS program without using an external tool. For example, here's a program that deletes an extraneous file from an existing ZIP file:

```
/* Remove the PackageMetadata piece that ODS PACKAGE creates */
filename pkg ZIP "c:\projects\filenamezip\new.zip" member="PackageMetaData";
data _null_;
 if (fexist('pkg')) then
  rc = fdelete('pkg');
run;
```

Note: Like ODS PACKAGE, the FILENAME ZIP method *does not* support encrypted (password-protected) ZIP archives.

Download the complete SAS 9.4 program: **filenameZipHttpExample.sas**

Thanks to the growing size of data files, ZIP files are created and consumed by SAS users everywhere. Between ODS PACKAGE and FILENAME ZIP, you can teach your SAS programs to build and read the files without having to rely on external tools. The more you that you can use native SAS methods for this work, the more portable your SAS programs will be.

See also
Using FILENAME ZIP to unzip and read SAS data files in SAS

tags: *FILENAME ZIP*, *PROC HTTP*, *SAS 9.4*, *ZIP files*

21 Comments

Jani Alatalo
Posted October 2, 2014 at 1:55 pm | *Permalink*

Beautiful!
Thanks for providing example how to get files within zip file listed!
I can use this immediately.

Reply

Mercedes
Posted December 2, 2015 at 2:26 pm | *Permalink*

But what if your external files are password protected how can you read from SAS directly ?
Thanks much

Reply

Chris Hemedinger
Posted December 2, 2015 at 2:31 pm | *Permalink*

For password-protected ZIP files, you'll still need to use external tools like

WinZip, 7-Zip, or gzip. This SAS Global Forum paper shows how that can work.

Reply

Manlio Fortuna

Posted November 14, 2014 at 6:35 am | *Permalink*

Hi Chris
I have a little problem because i receive a archive zip file that has the following structure
ZIP name
Nielsen
dir001 with file aaa
dir002 with file aaa
dir003 with file aaa
dir004 with file aaa
dir005 with file aaa
dir006 with file aaa
in each sub-folder there is a file that has the same name in all subdirectories but changes between reception of the zip file and the next and also the number of sub folder was variable.
how can i read all files present in archive zip file (the program is a automatic program without human control).

Reply

Chris Hemedinger

Posted November 20, 2014 at 4:57 pm | *Permalink*

If I understand you, it sounds like you have repeating file names within the same archive. In the archive they are in a folder structure, so they can be treated distinctly. But when you use a SAS program to process, all of the files end up in a single folder? You would need to rename the file as you extract it, perhaps based on the folder name, to keep the name unique in the folder. You could use the SAS RENAME function for this.

Reply

PGStats

Posted April 23, 2015 at 10:10 pm | *Permalink*

Hi Chris,
I met a similar situation. I couldn't find how to navigate a directory structure stored inside a zip file.

Reply

Marc Schlessel
Posted February 11, 2015 at 9:32 am | *Permalink*

Chris... Is there a way to read in a SAS dataset that's been zipped?

Reply

Chris Hemedinger
Posted February 13, 2015 at 4:05 pm | *Permalink*

Marc, yes, I think so. First, you would have to use the FILENAME ZIP method to copy the zipped data file from a ZIP archive. Then, you would assign a library to the location where you just copied that data file, and access the data from there.
You can do something like this to reach the file inside the ZIP, then copy it to a target folder:

```
filename _bcin zip "c:\temp\instanttitles.zip" member="instanttit
filename _bcout  "c:\projects\instanttitles.sas7bdat" recfm=n;
```

Then use something like the binaryFileCopy macro (which I shared in this post) to copy the file and access as data:

```
%binaryFileCopy()
%put NOTE: _bcrc=&_bcrc;

filename _bcin clear;
filename _bcout clear;

libname project "c:\projects";
proc datasets lib=project;
contents data=instanttitles;
```

```
quit;
```

Reply

Kevan Mather

Posted April 24, 2015 at 5:06 pm | *Permalink*

Hi Chris,
Thanks for this it's very useful!
What if there are excel files in the zipped folder, how would one go about reading them into SAS?
Kind Regards

Reply

Chris Hemedinger

Posted May 7, 2015 at 1:45 pm | *Permalink*

You can discover the member name using the example I provided here. Once you know the member name, you can assign a fileref to the Excel file you want to read. You can't PROC IMPORT the Excel file directly from the ZIP file, so you'll need to copy it out first. Here's an example that copies an XLSX file to the SAS Work location, and then runs PROC IMPORT on the result.

```
filename xl "%sysfunc(getoption(work))/sas_tech_talks_15.xlsx" ;
data _null_;
  infile inzip(sas_tech_talks_15.xlsx) recfm=n;
  file xl;
  input;
  put _infile_;
run;

proc import datafile=xl dbms=xlsx out=confirmed;
  sheet=confirmed;
run;
```

Reply

Darren Mayne
Posted June 9, 2015 at 10:00 am | *Permalink*

Hi Chris
Thanks for this, it was exactly the solution I was looking for to support an analytic project with zipped source files. Just out of interest, does the ZIP engine work with the newer ZIPX file format?
Cheers
Darren

Reply

Chris Hemedinger
Posted June 9, 2015 at 10:13 am | *Permalink*

Darren, I'm going to say that most likely: No, the FILENAME ZIP method would not support ZIPX (which I had to look up). That's a proprietary set of extensions created by WinZip.

Reply

Darren Mayne
Posted December 23, 2015 at 6:42 pm | *Permalink*

Thanks Chris.

Reply

José Bonilla
Posted September 26, 2015 at 10:50 am | *Permalink*

Thanks Chris, is there an equivalent for gzip (.gz) files?

Reply

Chris Hemedinger
Posted September 26, 2015 at 4:44 pm | *Permalink*

José,

gz is usually for a single file, while ZIP bundles up a collection of files and compresses them. There isn't a FILENAME method for gz, but ZIP and GZIP are mostly compatible. That is, you should be able to read a GZIPped file with the FILENAME ZIP method and vice versa. As some others have pointed out, password-protected files are not supported -- so that's one feature difference.

Reply

Peter Zajonc

Posted October 20, 2015 at 4:38 pm | *Permalink*

My data is encrypted and requires a password.
Can I add to the parameters? It is a simple fixed length file, once I give a pw, unzip it, and read it in!
Thanks

Reply

Chris Hemedinger

Posted October 20, 2015 at 9:24 pm | *Permalink*

Do you mean that the ZIP file is encrypted with a password? Then no, that is not supported by FILENAME ZIP.
If the ZIP file contains a SAS data set that is protected with a data set password, you can specify that in syntax when you read the data set. First, you must extract the data to a folder, then assign a library. This example shows you how.

Reply

Ravindranath

Posted January 8, 2016 at 1:50 am | *Permalink*

I have a folder, and i want to zip this folder by using SAS, create a password for zip folder by using sas.

Reply

Chris Hemedinger

Posted January 8, 2016 at 9:12 am | *Permalink*

Unfortunately, none of the SAS language methods (ODS PACKAGE, FILENAME ZIP) support passwords. You'll have to use the "old school" method: use X command or SYSTASK to call 7zip or gzip commands to compress with a password.

Reply

Siva

Posted February 24, 2016 at 11:28 am | *Permalink*

Hi Chris,
Thanks for the post. It's very helpful.

Reply

Chris Hemedinger

Posted February 24, 2016 at 11:36 am | *Permalink*

Glad to hear it! Thanks!

Reply

Post a Comment

Your email is *never* published nor shared. Required fields are marked *

Name *

Email *

Website

Comment *

You may use these HTML tags and attributes: <a href="" title=""> <abbr title=""> <acronym title=""> <b> <blockquote cite=""> <cite> <code> <del datetime=""> <i> <q cite=""> <s> <strike>

POST COMMENT

discovery, design of experiments, predictive modeling and more

SAS Learning Post
Technical tips and tricks from SAS instructors, authors and other SAS experts.

Graphically Speaking
Data Visualization with a focus on SAS ODS Graphics

SAS Users
Providing technical tips and support information, written for and by SAS users.

The Data Roundtable
A community of data management experts

COUNTRIES AND REGIONS

Hidden Insights
Experience the possibilities with Business Analytics in the Nordics

Klog på SAS
Tips og tricks til effektiv SAS-programmering

Mehr Wissen
Big Data Analytics in Deutschland, Österreich und der Schweiz

SAS Latin America
Business Analytics: Transforme su Negocio: Convierta sus datos en Conocimiento y Decisiones Inteligentes.

State and Local Connection

State and local governments using data to serve citizens and save money

The Analytic Hospitality Executive
Finding analytic solutions to your forecasting, pricing and operational challenges.

Pathfinders
Exploring SAS Curriculum Pathways & Instructional Technology

Made in… Analytics
Your guide for improving manufacturing outcomes with advanced analytics

---