Optimal large package administration for Stata

Markus Hahn

Rheinisch-Westfälisches Institut für Wirtschaftsforschung (RWI Essen)

March 28, 2006

Starting Point: "SOEP Menu"

- Easy and intuitive tool for data retrievals using the SOEP
  - German Socio-Economic Panel (SOEP) at the DIW Berlin
- Produces rectangularized data files directly into Stata's long format (use-ing, merging, reshaping, ...)
- Plugins (*.ado) ensure consistency over time
- Project files can be freely distributed: no data security issues
- SOEP Menu users donate €10/$10/£5 to UNICEF

- http://www.soepmenu.de

RWI ESSEN

"SOEP Menu" Distribution

- However the distribution is *large*

Package content

- Nearly 3800 files (author-written, user-written)
- Mostly text files (ado-files, html-files)
- Approx. 47 MBytes (uncompressed)

Updates

- Every year: New version ready for the newest SOEP Distribution
- Major and minor updates during the year

Previous Method of distribution

- *One* encrypted zip file via internet (download and unzip by hand)

RWI ESSEN

Drawbacks of the *old* method

- Old is bad (a law of nature)
- Awkward installation and update process
- Even minor updates require the same effort as major updates

A New Hope

- IDEA: There should be an "automatic" update function
- Something like Stata's update function
- This update function could also be used as a new installation routine

RWI ESSEN

The Download Servers Strike Back

Attempt One:

- Download of all files separately (like Stata's update)
- The files are uncompressed: Produces much download traffic
- ⇒ Result: Duration 24min, time-outs possible

Attempt Two:

- Download of all files separately
- The files are now compressed with `gzip`: Less download traffic
- ⇒ Result: Duration 20min, time-outs possible

Cause for the low performance:
⇒ There are 3800 download queries, one for each file

RWI ESSEN

Attempt Three: smnetupdate.ado (Update for SOEP Menu)

- Only three major `tar`-archives zipped with `gzip`
- The files are encrypted with `ccrypt`
- Compares local index files with the server index file
- Archives are downloaded respectively if necessary
- $\Rightarrow$ Result: Duration $<$ 3min

- Whole installation routine
  `smnetupdate, install(c:/soepmenu)`
- Also an Administrative Tool to create Package Distributions
  - New update packages can easily and "automatically" be created

RWI ESSEN

Attempt Three: How the comparision is done

Local version	Server version	
`16Feb2006151310`	`21Mar2006112049`	$\Rightarrow$ Update necessary

Possible Extensions

- Can easily extend to additional directories or files
- Check total archive (3) or check each individual file (3800)?
- Trade-off: longer start-up times due to checking

RWI ESSEN

Problems with our implementation so far

- `tar`, `gzip` and `ccrypt` are needed as additional binaries
  - Automatic download on windows machines
  - Other operation systems most likely need `ccrypt` installed
- Execution via shell: `shell tar...`
  $\Rightarrow$ Confusing command-shell popups in Windows

- Therefore we would like to suggest some possible improvement to Stata which deal with these issues...

RWI ESSEN

Suggestion: Compression

- Compressed data files
  `use var1 var2 using data.zip`
- Compressed files
  `file open fh using compressed_file, text` *`compressed`*
- `unzip anyfile` / `zip anyfile`
- Saves disk space (80-90%) and network traffic
- Usefull for backups

Suggestion: Encryption

- Encrypted variables
  `generate secretvar = encrypt(var, "PASSWORD")`
  - Some variables only available to persons with proper authentification
  - for example: GIS, sensitive zip code, other individual information
- Encrypted data files
  `use secret_datafile, password("PASSWORD")`
- `encrypt anyfile, password("PASSWORD")`
  $\Rightarrow$ Encrypted log files, ...

RWI ESSEN

Suggestion: Expanded Package file (.pkg)

- A PKG administration like `tar`
- Individual files are packaged together into one archive file
- This package could also be compressed
- Descriptive information like Stata's current implementation of pkg-files

Conclusion

- Query time: Minimize number of download queries
- Download volume: Minimize volume of downloads
- Fewest possible queries, lowest possible download volume