

## Running Rapp and editing Rapp programs

2018-08-17, Stig Rosenlund

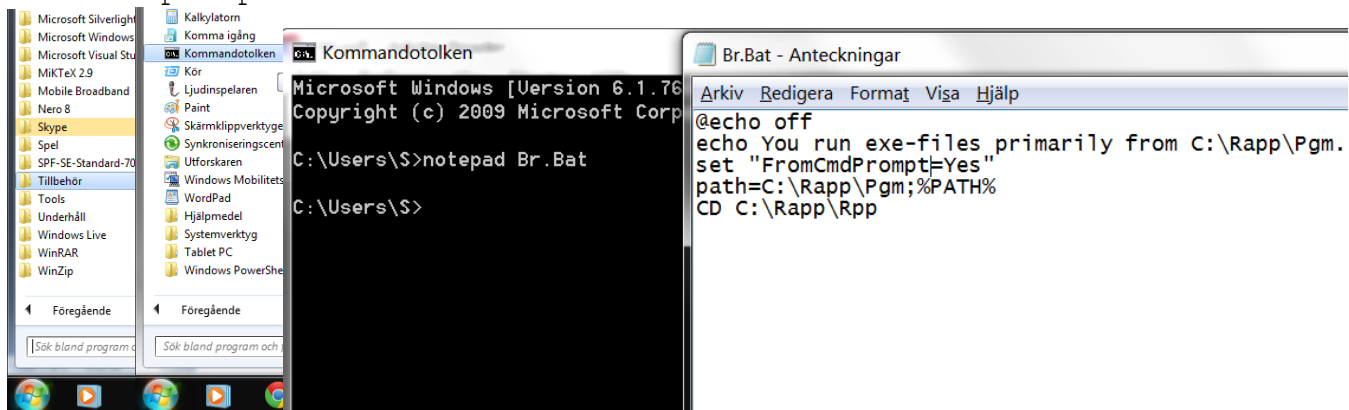
We assume you have created the folder C:\Rapp from the downloaded zipfile Rapp.zip or Rapp.zipx.

### 1. Run and edit Rapp.Exe without using Rappmenus.Exe

#### 1.A. Running Rapp from the command prompt

Initiation: create an environment at the command prompt  
Startmenu / All programs / Accessories / Command prompt

I illustrate how to do it below. In Swedish Accessories is Tillbehör and Command prompt is Kommandotolken.



Copy the five lines below.

```
@echo off
echo You run exe-files primarily from C:\Rapp\Pgm.
set "FromCmdPrompt=Yes"
path=C:\Rapp\Pgm;%PATH%
CD C:\Rapp\Rpp
```

At the prompt >, write

Notepad Br.Bat

Answer Yes to Create new file?

Paste the five lines into new file Br.Bat. Close up right and save.

Place a shortcut to Command prompt on the desktop or the Startmenu.

#### Daily use

When you are going to run a Rapp program, go into the Command prompt and write Br and press Enter.

For example there is a simple program C:\Rapp\Rpp\HelloWorld.Rpp. Run it at the Command prompt by

Rapp HelloWorld

### 1.B. Running Rapp from Windows Explorer

Initiation: Associate extent .Rpp to Rapp.Exe in Windows Explorer  
 Either run Adapt Exe or

Click or rightclick in Windows Explorer on a Rapp-program with extent .Rpp (for example Init.Rpp), select Open / Choose Program / Browse / locate Rapp.Exe and tick Always use this program.

#### Daily use

Run a Rapp-program by doubleclick or click+return in Windows Explorer. When its done, press enter to exit Rapp.

### **Advantages and disadvantages of running Rapp from the command prompt**

#### Advantages

1. It will always work, while the association in Windows Explorer might get lost and be difficult to restore due to authorization problems.
2. The working directory will always be C:\Rapp\Rpp. Some Rapp programs presume this. In Windows Explorer it might be something else.
3. As Rapp developer, I can make a sequence of versions of Rapp for testing some new feature, and run them quickly. For example I might make t1.exe, which is Rapp with some change, and run it in the command prompt with t1 tst1. That is inconvenient in Windows Explorer.

#### Disadvantages

1. It takes some time to write a long Rapp program name.
2. It has a feeling of being old and not elegant.

Disadvantage no 2 is just superficial, though.

### 1.C. Editing Rapp programs

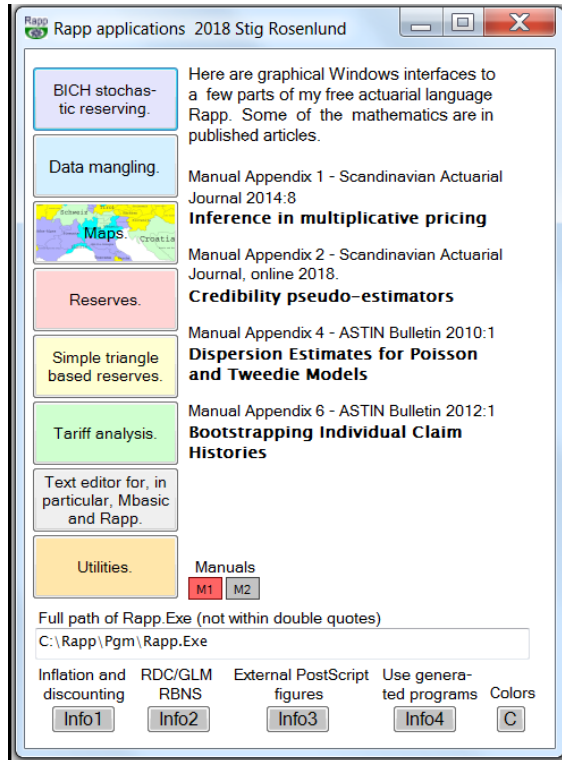
To write Rapp-programs you can use a text editor you're used to, such as the SAS source editor, Wordpad or Notepad. I recommend SPF Source Editor from CTC Command Technology Corporation. It costs some money, presently \$88 for perpetual use. Web site:

<http://www.commandtechnology.com>

### **2. Run and edit in Rappmenus.Exe.**

Besides these ways there is Rappmenus.Exe, which generates and runs Rapp programs for certain applications dynamically. In 2017 and 2018 it has been amended with a text editor, suitable for any text file. Rapp programs can be executed with a Run button. SPF has more capabilities, but takes some time to learn.

In Rappmenus, choose "Text editor for, in particular, Mbasic and Rapp." in the menu below.



The editor has coloring of special syntax words in Mbasic (part of Rapp, see Proc Mbasic in the manual) and Rapp in general. There is also syntax-sensitive coloring for C and for some of the keywords of SAS (those I used at Länsförsäkringar). I have emulated some of the functionality of SPF. An Mbasic- or general Rapp-program is run from the editor with a button. There are some useful template parmfiles to get you started in the folder Mbparm of the zip files Rapp.zip and Rapp.zipx. Place this folder on your computer as C:\Rapp\Mbparm. Find, replace and sort are extensively implemented. There are special features which facilitate debugging your Mbasic-program.

Below an example where an Mbasic-program is in the edit screen.

```

000001 /* Computes Euler constant 0.5772156649015... . Computation time
000002 increases rapidly with n, but n <= 15 will not take too long.
000003 Due to rounding errors the last few (about six) digits will be wrong.
000004 If you want k correct digits, set digits to k + 9 or so. */
000005 digits 306 // Maximum with Rapp - larger maximum with Rapp1008 etc.
000006 double m n
000007 mouble Hm One Sum Sumold Two Two raised to n e raised to Two raised to n m01
000008 Read n // The computation will have a remainder term O(1/(2**n * exp(2**n))).
000009 One = 1
000010 Two = 2 Sum = 1
000011 Hm = 1
000012 m01 = 1
000013 Two raised to n = Two**n
000014 e raised to Two raised to n = exp(Two raised to n)
000015 For m=1 to 999999999999999 // Infinite loop interrupted by condition below.
000016 // Mouble variable One is needed for mouble precision.
000017 Hm = Hm + One/(m+1) // Harmonic series 1 + 1/2 + 1/3 + 1/4 + 1/5 + ...
000018 m01 = m01*Two raised to n/(m+1)
000019 Sum = Sum + m01*Hm
000020 if Sum = Sumold then break endif // When the addition of m01*Hm is too small.
000021 Sumold = Sum
000022 Next m
000023 // log(Two) is needed since log(2) will yield precision only double.
000024 Sum = Sum*Two raised to n/e raised to Two raised to n - n*log(Two)
000025 print "Approximate value of Euler's constant is:" \n,%1.300 Sum \n
000026 print to "C:\Rapp\Txt\Euler.Txt"
000027 print %1.300 Sum \n
000028 print to
000029 close "C:\Rapp\Txt\Euler.Txt"
000030 system "C:\Rapp\Txt\Euler.Txt" // See the result in Notepad.
000031
000032
000033

```

Another Mbasic-program. Several parts have been joined.

```

000001 /* The following program performs portfolio optimization. It is very simple.
000002 I assume that programs made by commercial financial software companies
000003 are much better. See Appendix 11 for the formulas used. */
000004 double E1 alfa1 alfa2
000005     F 4 G 4 H 4 H1_4[1][4] H2_4[1][4] V 4[4][4] Vinv_4[4][4]
000006     c 4[4][1] e 4[4] etr 4[1][4] m01_4 mu 4[4] mutr 4[1][4]
000007     F 3 G 3 H 3 H1_3[1][3] H2_3[1][3] V 3[3][3] Vinv_3[3][3]
000008     c 3[3][1] e 3[3] etr 3[1][3] m01_3 mu 3[3] mutr 3[1][3]
000009
000010 e 4 = { 1 1 1 1 }
000011 mu 4 = { 1.100 1.100 1.175 0.850 }
000012 v 4 = {
000013     -0.030000 0.030000 0.030000 -0.030000
000014     0.030000 0.060000 0.030000 -0.030000
000015     0.030000 0.030000 0.076875 -0.030000
000016     0.030000 -0.030000 -0.030000 0.037500
000017 }
000018 etr 4 = trp(e 4)
000019 mutr 4 = trp(mu 4)
000020 Vinv 4 = invsp(V 4)
000021 H1 4 = etr 4*Vinv 4
000022 H2 4 = mutr 4*Vinv 4
000023 F 4 = H1 4*e 4
000024 G 4 = H1 4*mu 4
000025 H 4 = H2 4*mu 4
000026 m01 4 = G 4*G 4 - F 4*H 4
000027
000028 e 3 = { 1 1 1 }
000029 mu 3 = { 1.100 1.100 1.175 }
000030 v 3 = {
000031     -0.030000 0.030000 0.030000
000032     0.030000 0.060000 0.030000
000033     0.030000 0.030000 0.076875
000034 }
000035 etr 3 = trp(e 3)
000036 mutr 3 = trp(mu 3)
000037 Vinv 3 = invsp(V 3)
000038 H1 3 = etr 3*Vinv 3
000039 H2 3 = mutr 3*Vinv 3
000040 F 3 = H1 3*e 3
000041 G 3 = H1 3*mu 3
000042 H 3 = H2 3*mu 3
000043 m01 3 = G 3*G 3 - F 3*H 3
000044
000045 print to "C:\Rapp\Txt\Portopt.Txt"
000046 print "          Standard" \n
000047 print "Expected deviation      Distribution percent" \n
000048 print "  yield      percent      Ass1  Ass2  Ass3  Ass4" \n
000049 for E1 = 1.005 to 1.175 by 0.005
000050     alfa1 = (G 4*E1 - H 4)/m01 4
000051     alfa2 = (G 4 - E1*F 4)/m01 4
000052     c 4 = Vinv 4*(alfa1*e 4+alfa2*mu 4)
000053     if not(c 4[1][1] >= 0 and c 4[2][1] >= 0 and c 4[3][1] >= 0 and c 4[4][1] >= 0) then
000054         // Asset no 4 is excluded. The one to exclude was simply determined by running
000055         // first without these conditional statements, which gave negative c 4[4][1].
000056         alfa1 = (G 3*E1 - H 3)/m01 3
000057         alfa2 = (G 3 - E1*F 3)/m01 3
000058         c 3 = Vinv 3*(alfa1*e 3+alfa2*mu 3)
000059         c 4 = c 3
000060     endif
000061     print %8.3 E1, %10.2 100*sqrt(trp(c 4)*V 4*c 4), " ", %6.2 100*trp(c 4)
000062 next E1
000063 print to
000064 close "C:\Rapp\Txt\Portopt.Txt" // The file must be closed before it can be viewed.
000065 system "C:\Rapp\Txt\Portopt.Txt"

```

A Rapp-Program for multivariate linear regression, in Overwrite mode.  
 "Hide line nos" has been checked to turn off line numbering.

```

include C:\Rapp\Rpp\Init.Rpp
/*
Proc Linreg performs regular multivariate linear regression, based
on H Cramér, Mathematical Methods of Statistics, Ch 37.3. Provides
point estimates, confidence intervals and p-values. Has no specific
mathematical features - use SAS for more advanced methods.
*/
Proc Linreg
  infil(C:\Rapp\Data\Linreg01.Txt) log level(95)
  utfil(C:\Rapp\Txt\Linregout.txt) gutfil(C:\Rapp\Txt\Lingraf.txt)
Endproc
Proc Graf listfil(C:\Rapp\Txt\Lingraf.txt) pdffil(C:\Rapp\Pdf\Lingraf.Pdf) Endproc
system(del C:\Rapp\Txt\Lingraf.txt & C:\Rapp\Pdf\Lingraf.Pdf)

```

Proc Gpdml for the generalized Pareto distribution.

```

include C:\Rapp\Rpp\Init.Rpp
Proc Gpdml Infil(C:\Rapp\Data\Catastro-claims.Txt) Utfil(C:\Rapp\Txt\C01.Txt) Endproc
// system(C:\Rapp\Txt\C01.Txt)
Proc Excel listfil(C:\Rapp\Txt\C01.Txt) Xmlfil(C:\Rapp\Xml\C01.Xml) Headerdoubleremove visa ENDPROC

```

A Rapp procedure outside the area of mathematical statistics.

```

include C:\Rapp\Rpp\Init.Rpp
// Produces a calendar for any year from 1582. Week numbers by
// ISO 8601 effective 1972. Default CalendarYear() is present year.
Proc Init lan(e) Endproc
// Proc Calend Pdffil(C:\Rapp\Pdf\a1.Pdf) CalendarYear(1889) charsperday(1) visa ENDPROC
Proc Calend Pdffil(C:\Rapp\Pdf\a2.Pdf) prompt visa charsperday(1) ENDPROC

```

All buttons and boxes are explained at right-click. The red button M displays the Rapp manual in pdf. The red button P next to it displays the text in edit in pdf. One can think of P as meaning Print, since the pdf file will give a good printout on paper with Ctrl+P.