Cobol + Tcl/Tk Interface with named pipes from Vaclav Snajdr

V.Snajdr

Logol 1974 (LogAbax, Sligos)

Cobol 1981 (Texas Instruments), 1985 Unix, 1998 Linux

Tcl/Tk 1999

More bookkeeper (ERP concept and programming) than IT-professional

Motivation 1: GUI for Cobol

History: First interface Tcl-Cobol – Hamburg 2000 – Franco Violi with C, Tcl-C, statically solution, formulars, not good interaction

Second interface 2005 – Wolfgang Grossbauer, Client/Server, with C, statically linking C with Cobol.

Servers: AS400, Windows, Linux. Clients: Windows, Linux.

Some Problems with changing libraries on diverse Linux-Versions (Suse, Ubuntu, Debian, etc.)

Third interface - 2009 – myself, based on "named pipes", no client/server, desktop X11 on Server, quick access with NX-Products(www.nomachine-com)

How it works: Tcl/Tk starts first, create pipes

```
set   a   CfrT;            #  cobol from tcl
set   f   $::VDT$a;        #  name of pipe-file included /dev/tty in VDT
exec  mkfifo    $f;        # create pipe
set ::ou [open $f w+]; # open pipe
set   a  CtoT;             # cobol to tcl
….                         # create …
set ::in [open $f r+]
exec CobolMain.sh $::VDT &;   # start cobol proces
```

Write into pipe:

```
proc putSock {datentk} {
        puts  $::ou $::TKDATEN
        flush $::ou
        }
```

Read from pipe:

```
proc getSock {} {
        if {[gets $::in record] <0} {return}
        set  record [string trimright $record \x00]
        set  ::TKDATEN $record
}
```

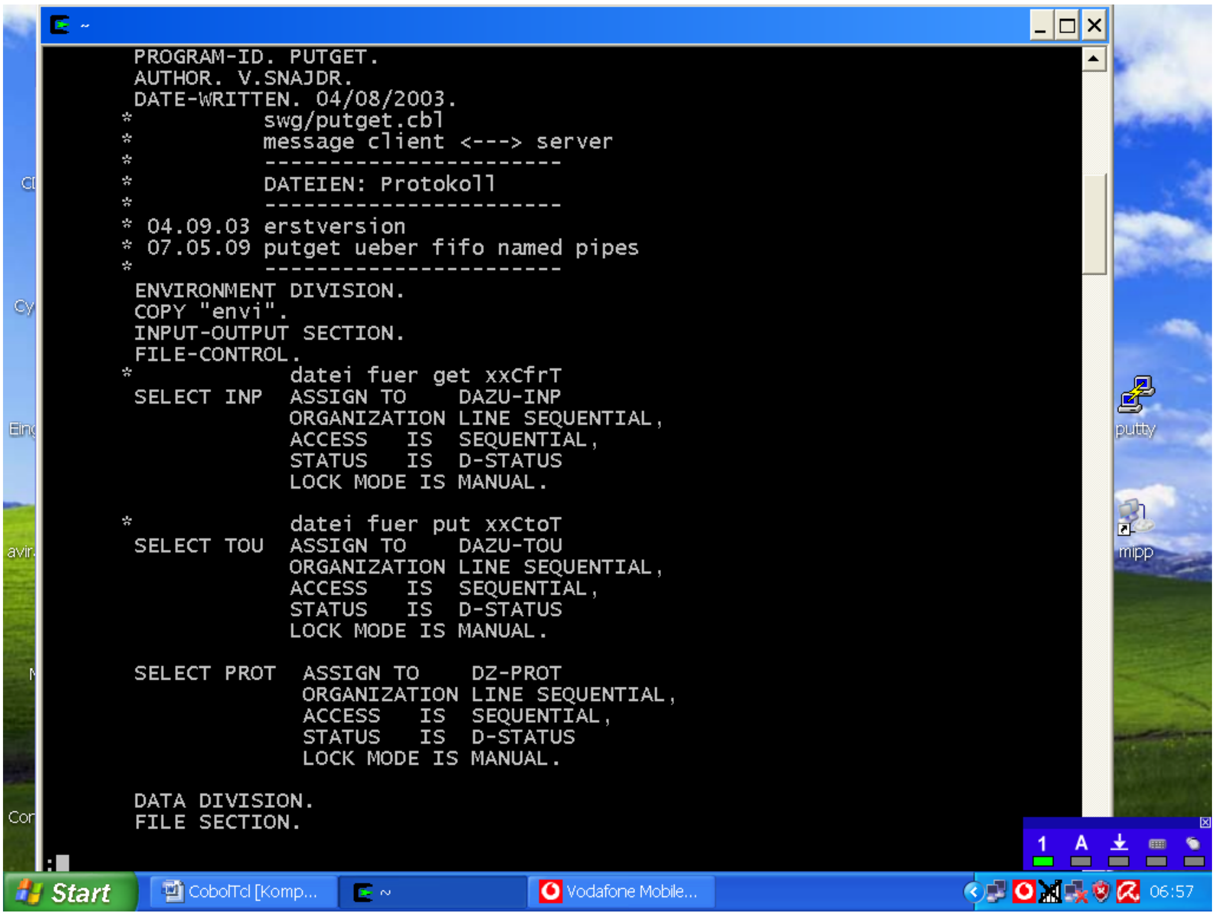
Write/Read:

```
proc pu_ge {} {
        putSock $::TKDATEN
        set      ::TKDATEN [getSock];
}
```

Example create new record - ::TKDATEN says what Cobol should do

```
proc write_dk {} {
          set aktion "aktion=neu_dk";
          set inhlog "kurz_dk=$::MYLOG"
          set inhpas "pasw_dk=$::MYPWD"
          set wekuli  "kuorli=$::KUORLI"
         set ::TKDATEN "RUPA:msdklog:$aktion:$inhlog:$inhpas:$wekuli:";
         pu_ge;
```

Cobol: two sequentile files for pipes, one protocol file



```
       PROGRAM-ID. PUTGET.
       AUTHOR. V.SNAJDR.
       DATE-WRITTEN. 04/08/2003.
      *          swg/putget.cbl
      *          message client <---> server
      *          -----------------------
      *          DATEIEN: Protokoll
      *          -----------------------
      * 04.09.03 erstversion
      * 07.05.09 putget ueber fifo named pipes
      *          -----------------------
       ENVIRONMENT DIVISION.
       COPY "envi".
       INPUT-OUTPUT SECTION.
       FILE-CONTROL.
      *            datei fuer get xxCfrT
       SELECT INP  ASSIGN TO    DAZU-INP
                   ORGANIZATION LINE SEQUENTIAL,
                   ACCESS    IS SEQUENTIAL,
                   STATUS    IS D-STATUS
                   LOCK MODE IS MANUAL.

      *            datei fuer put xxCtoT
       SELECT TOU  ASSIGN TO    DAZU-TOU
                   ORGANIZATION LINE SEQUENTIAL,
                   ACCESS    IS SEQUENTIAL,
                   STATUS    IS D-STATUS
                   LOCK MODE IS MANUAL.

       SELECT PROT  ASSIGN TO    DZ-PROT
                    ORGANIZATION LINE SEQUENTIAL,
                    ACCESS    IS SEQUENTIAL,
                    STATUS    IS D-STATUS
                    LOCK MODE IS MANUAL.

       DATA DIVISION.
       FILE SECTION.
:
```

Cobol record size is here max. 64 KB.

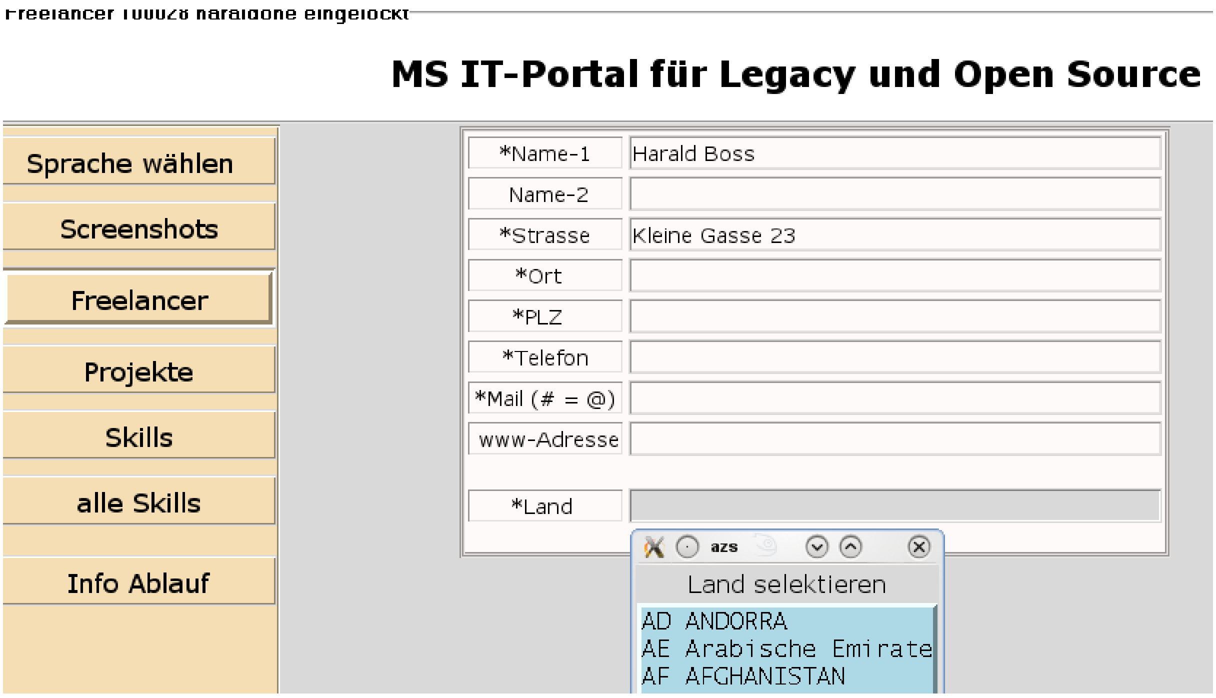
Write and read the pipes is similar to Tcl.

Motivation 2 for using Cobol: I have a complet ERP (Enterprise Ressource Planing) software package with about 800 Moduls and 70 Datatables. To use the existing routines ist better than to rewrite the same in another language only.

Live demo: A new person will be created

- GUI Tcl/Tk – input on screen with some control
- Cobol on server takes the data and write into database

Picture 1: Screen input with Tcl/Tk



Freelancer 100028 haraldone eingelockt

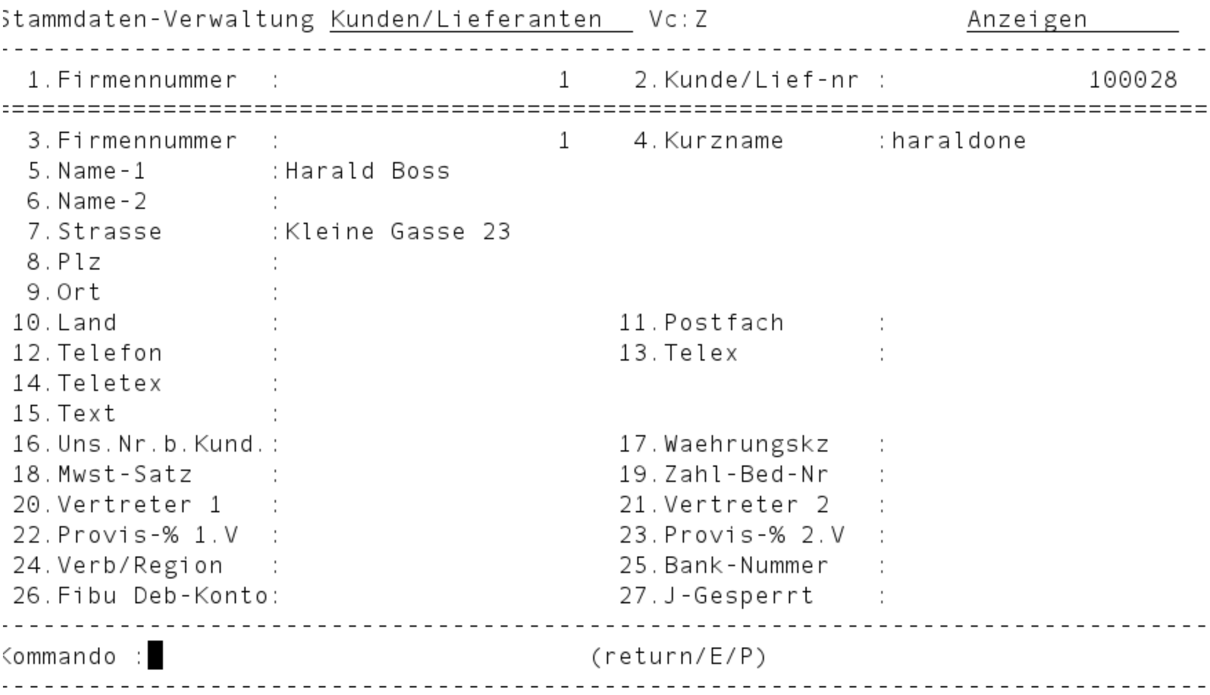
MS IT-Portal für Legacy und Open Source

*Name-1	Harald Boss
Name-2	
*Strasse	Kleine Gasse 23
*Ort	
*PLZ	
*Telefon	
*Mail (# = @)	
www-Adresse	
*Land	

Land selektieren

AD ANDORRA
AE Arabische Emirate
AF AFGHANISTAN

Picture 2: Every filed will be immediatily send to Cobol and write into database



```
Stammdaten-Verwaltung Kunden/Lieferanten   Vc:Z                  Anzeigen
-------------------------------------------------------------------------------
  1.Firmennummer    :                  1    2.Kunde/Lief-nr  :           100028
===============================================================================
  3.Firmennummer    :                  1    4.Kurzname      :haraldone
  5.Name-1          :Harald Boss
  6.Name-2          :
  7.Strasse         :Kleine Gasse 23
  8.Plz             :
  9.Ort             :
 10.Land            :                      11.Postfach      :
 12.Telefon         :                      13.Telex         :
 14.Teletex         :
 15.Text            :
 16.Uns.Nr.b.Kund.  :                      17.Waehrungskz   :
 18.Mwst-Satz       :                      19.Zahl-Bed-Nr   :
 20.Vertreter 1     :                      21.Vertreter 2   :
 22.Provis-% 1.V    :                      23.Provis-% 2.V  :
 24.Verb/Region     :                      25.Bank-Nummer   :
 26.Fibu Deb-Konto  :                      27.J-Gesperrt    :
-------------------------------------------------------------------------------
Kommando :█                             (return/E/P)
-------------------------------------------------------------------------------
```

Very usefull for control during the developing phase.

Strasbourg, 5.6.2010