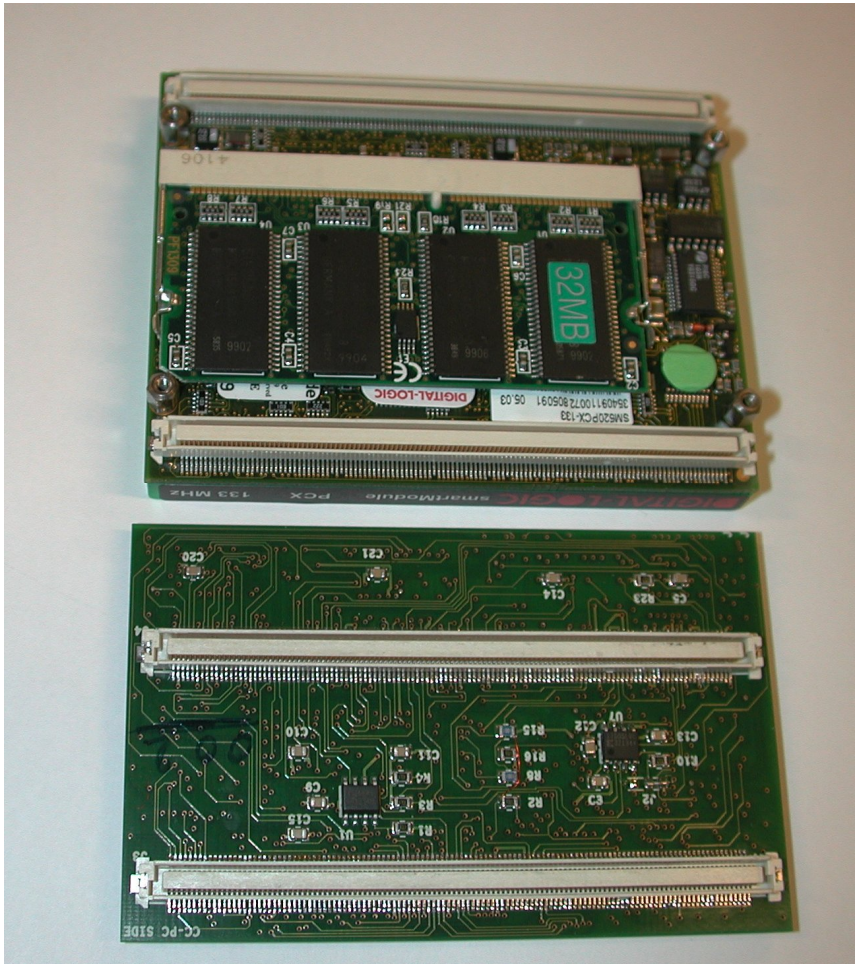
The Credit Card PC:

The Server:

- New Release
- Outlook

Tools (PVSS):

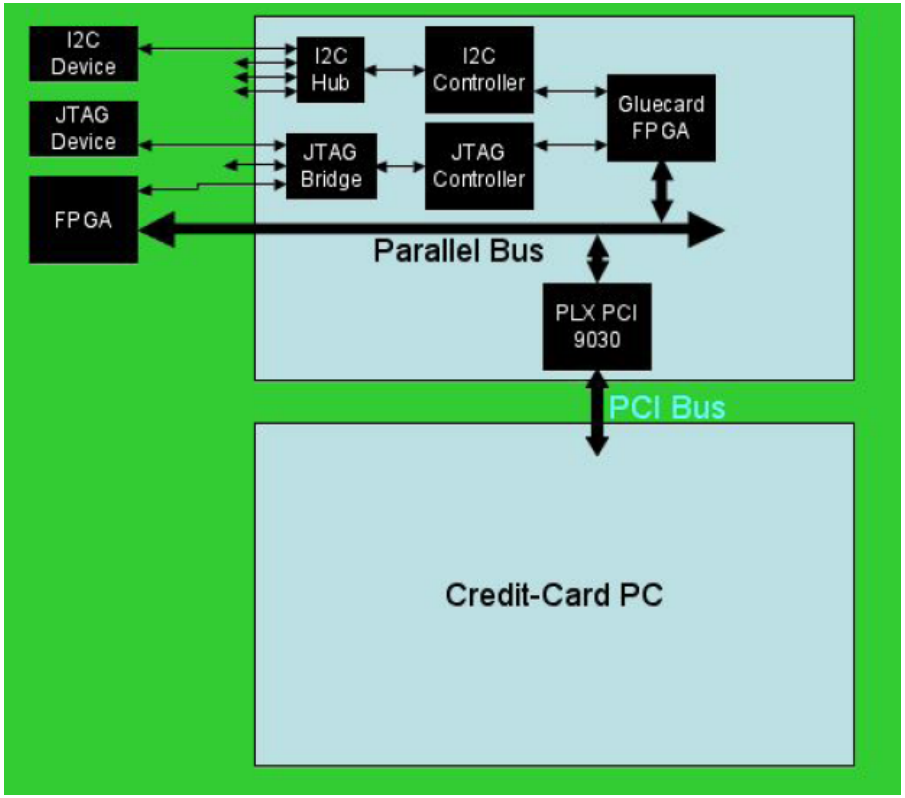
- a brief introduction



Introduction (I):

Plx/PCI bridge to gluecard

3 bus-types:
- **parallel bus (8/16/32 bits)**
- **I2C bus (4 distinct lines)**
- **JTAG (3 chains)**

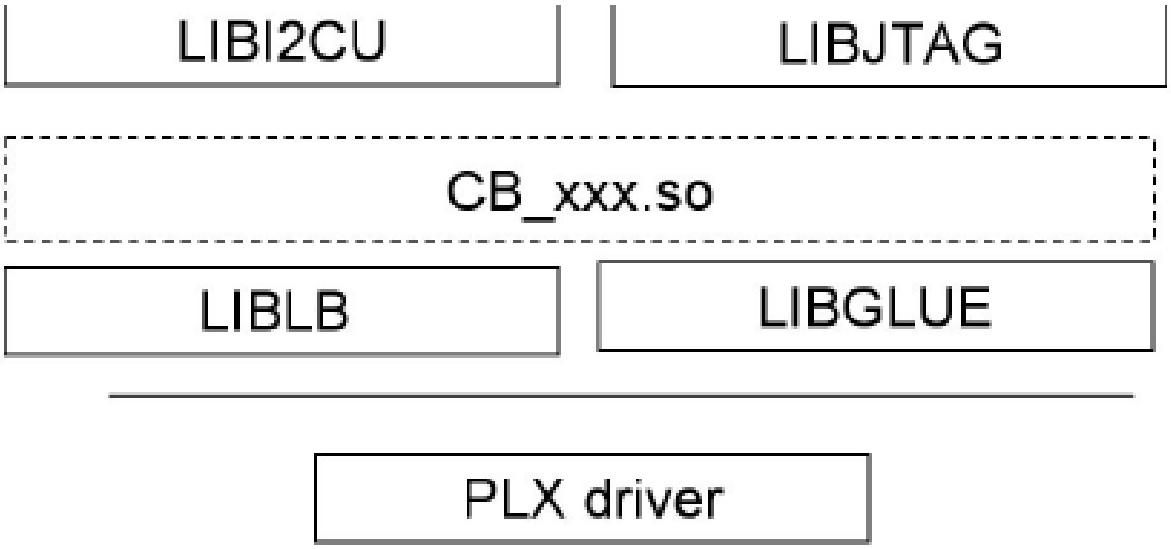


CCPC is used on:
- **Tell1 board**
- **UKL1 board (16 bit local bus)**
- **TFC boards**
- **OPG and others (L0)**

Different Hardware but same server!

Introduction (II):

The software layers on which the server is based on:



These libraries are very reliable and well tested.
Currently running under SLC4.

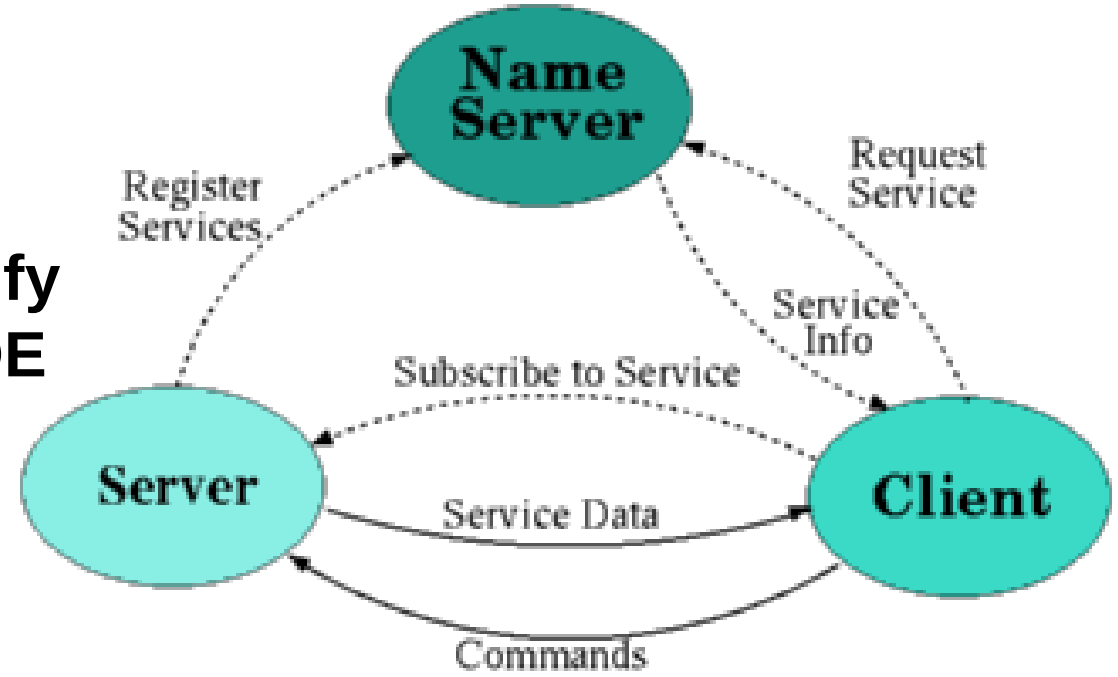
More Information:

Niko Neufeld:
The CCPC Software developers guide

http://lhcb-daq.web.cern.ch/lhcb-daq/ccpc/development/guide.pdf

Introduction (III):

Communication between server and clients based on DIM:



Don’t forget to specify your DIM_DNS_NODE (and have one running)!

The PVSS clients are presented later in more detail …

More Information:

Clara Gaspar:
Distributed Information Management System

http://dim.web.cern.ch/dim

Server (I):

- In order to have similar framework functions on the PVSS client-side for both SPECS & CCPC the interface has changed.

- The way 'dpSet' is implemented in PVSS seems to affect the performance:
The new interface tries to optimize this.

- Traffic is kept as low as possible:
All register information is stored in lists on the server-side. On the other hand, registers are just updated on the client-side if they have really changed.

Server (II):

- A simple interface shall be provided on the user-side:
try as much as possible to hide complexity of the different hardware types.
(for the benefit of the user and the pain of the developer J)

- Future release will add functionalities whithout breaking existing PVSS interfaces.
(user code is safe)

- Existing SPECS panels adapted for the CCPC. (Ricardo)

Server (III):

Future tasks:

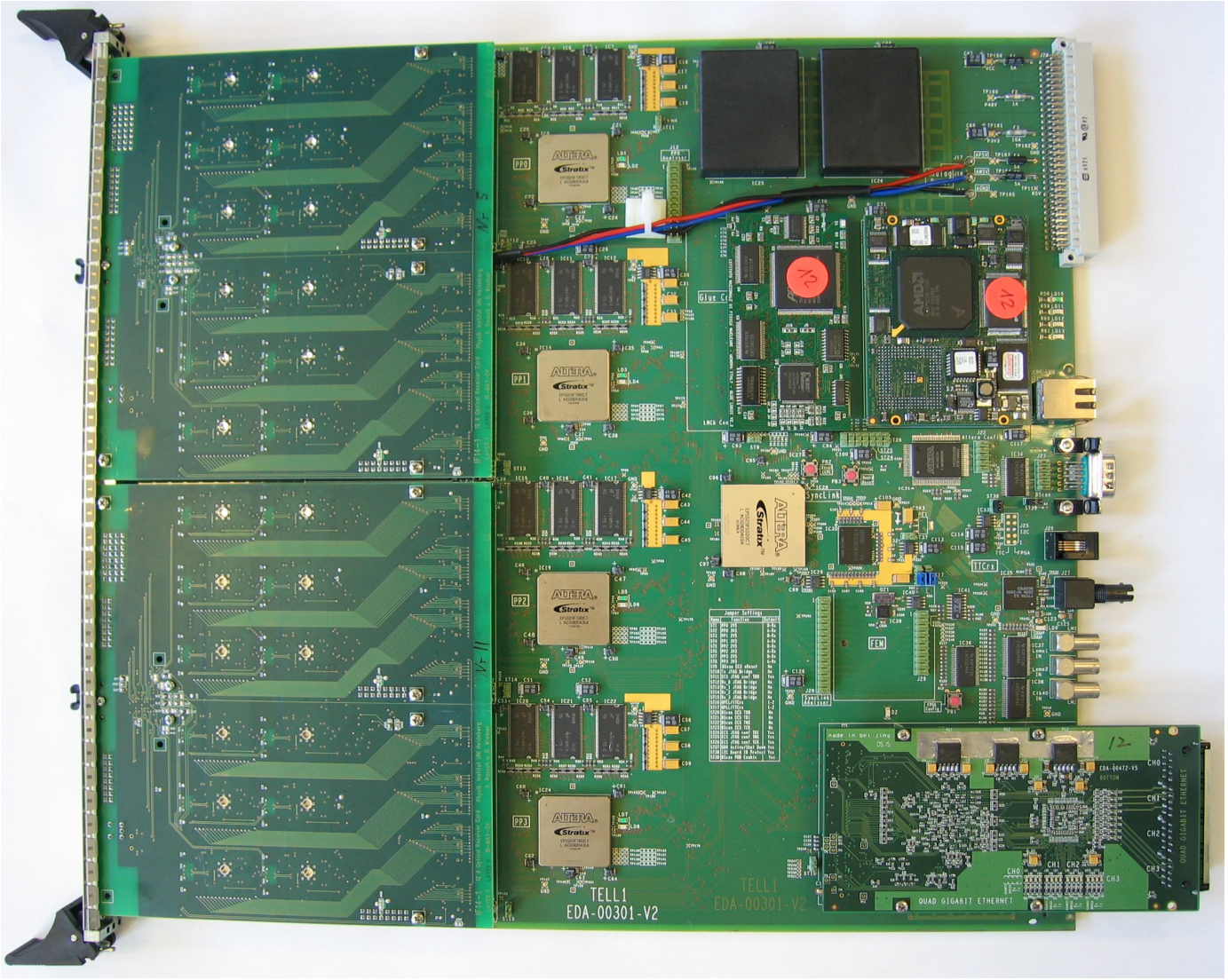
- Model the whole Tell1 board (using the new tools) as PVSS datapoint types.

- Defining a higher level framework for the Tell1 to be used by finite state machines or modelling macroscopic functions (e.g. exploiting the powerful PHY chip on the GBE-card, etc).

- Monitoring of registers bundled to further improve performance.

- Connection to configuration database. Up- and Download of default values. (defining PVSS recipies)

Use-case Tell1 board (I):

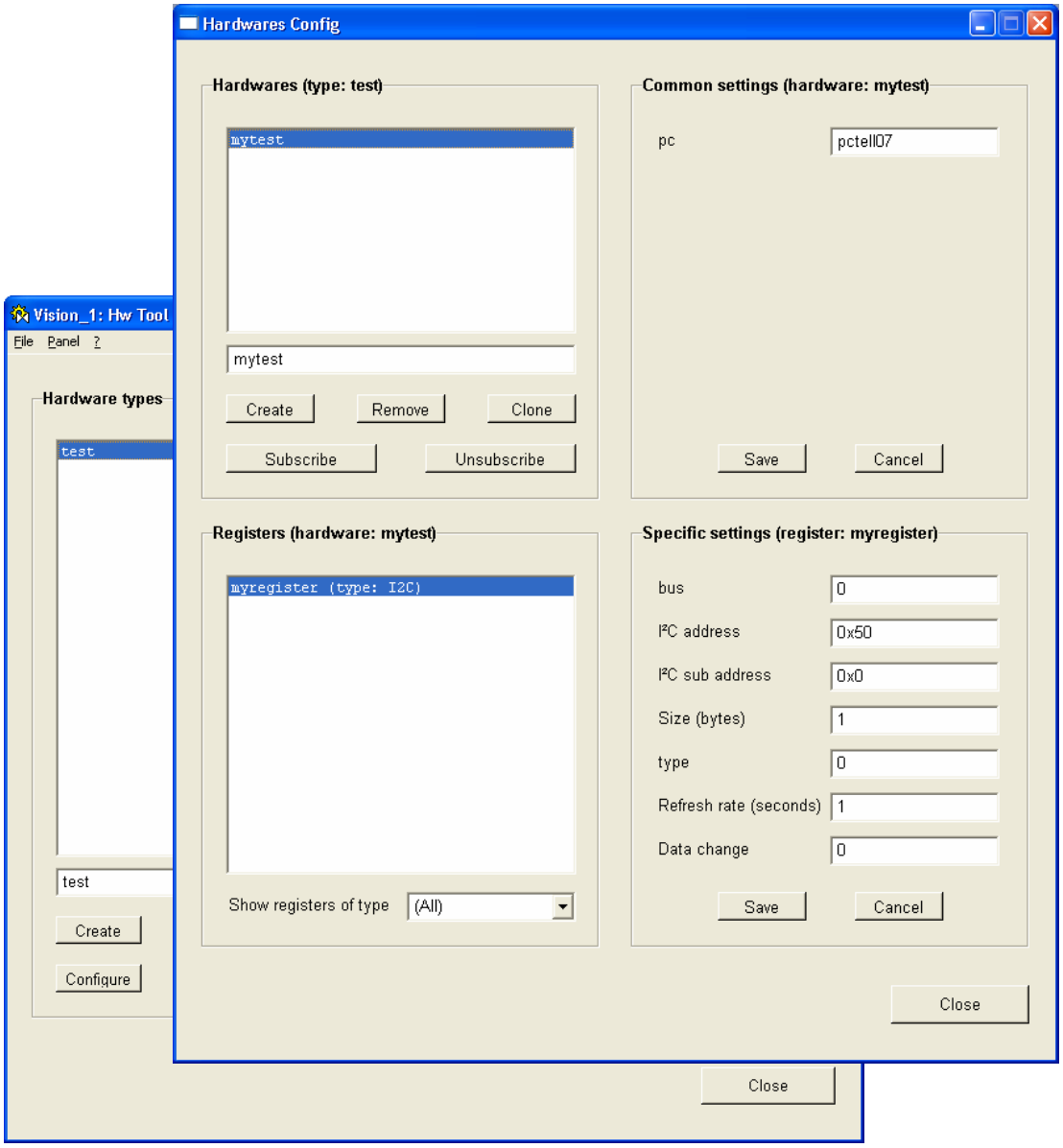


Use-case Tell1 board (II):

Creating register-types with FwHw-tool (e.g. I2C):

Similar to the specs-system. (as in tutorial yesterday)

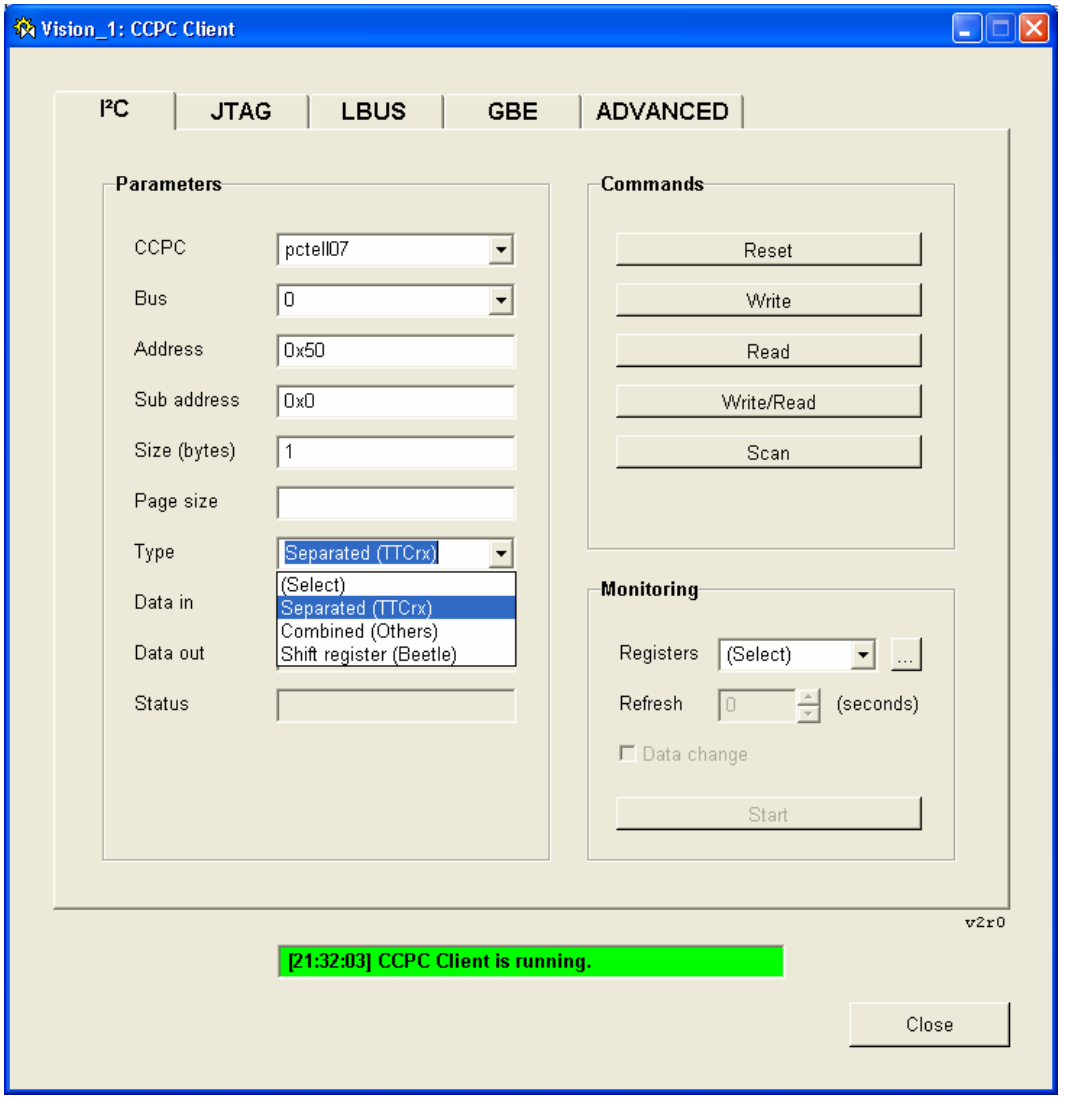
Parameters to be defined are however different! I2C,LBUS,GBE provided so far.



Use-case Tell1 board (III):

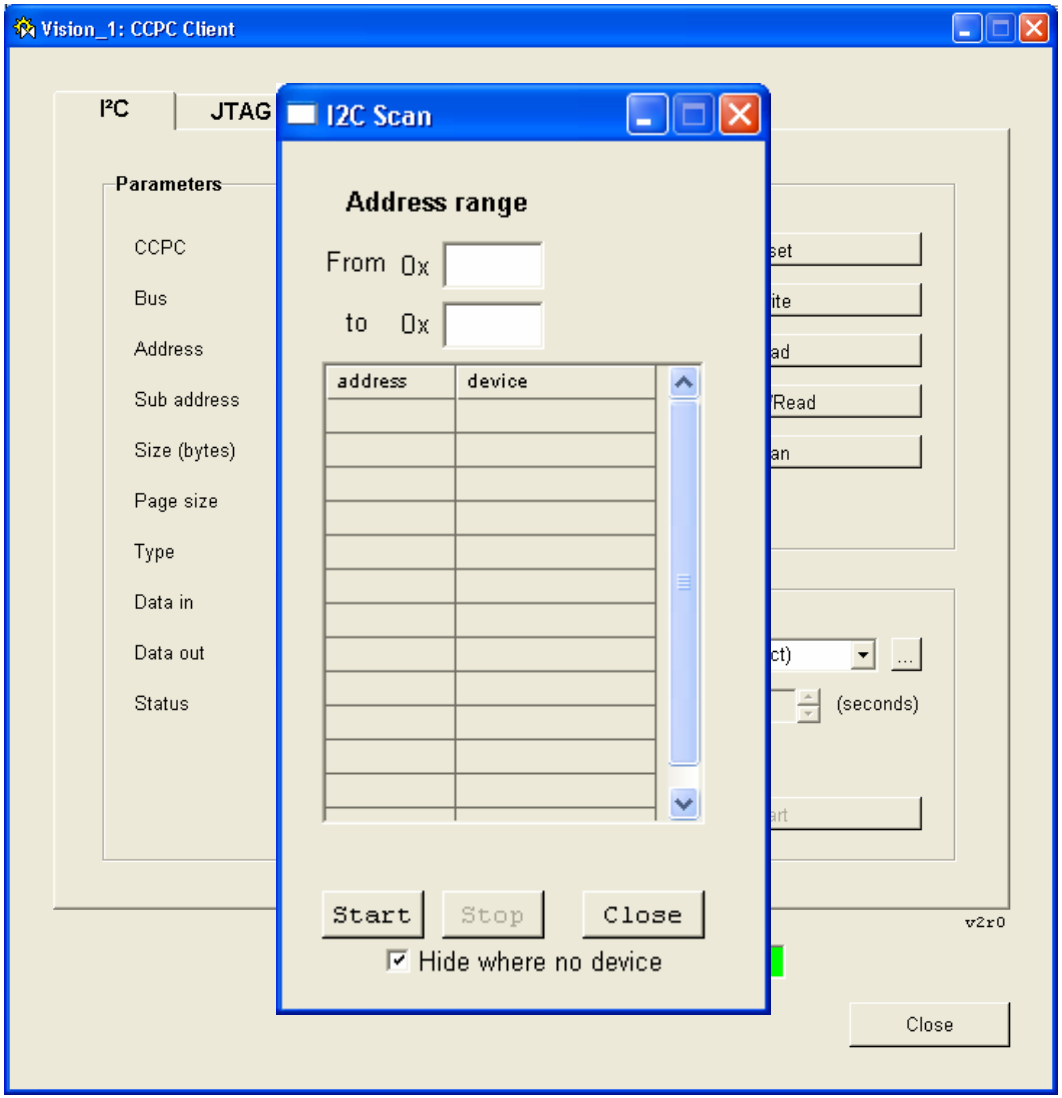
R/W or Monitoring (e.g. I2C):
again similar to the tutorial yesterday!

Or not so similar:
TTCrx has non-standard I2C interface!
The ID-EEPROM allows just a 16 Byte write access (pagesize)!
Beetle offers 8bit shift registers ….



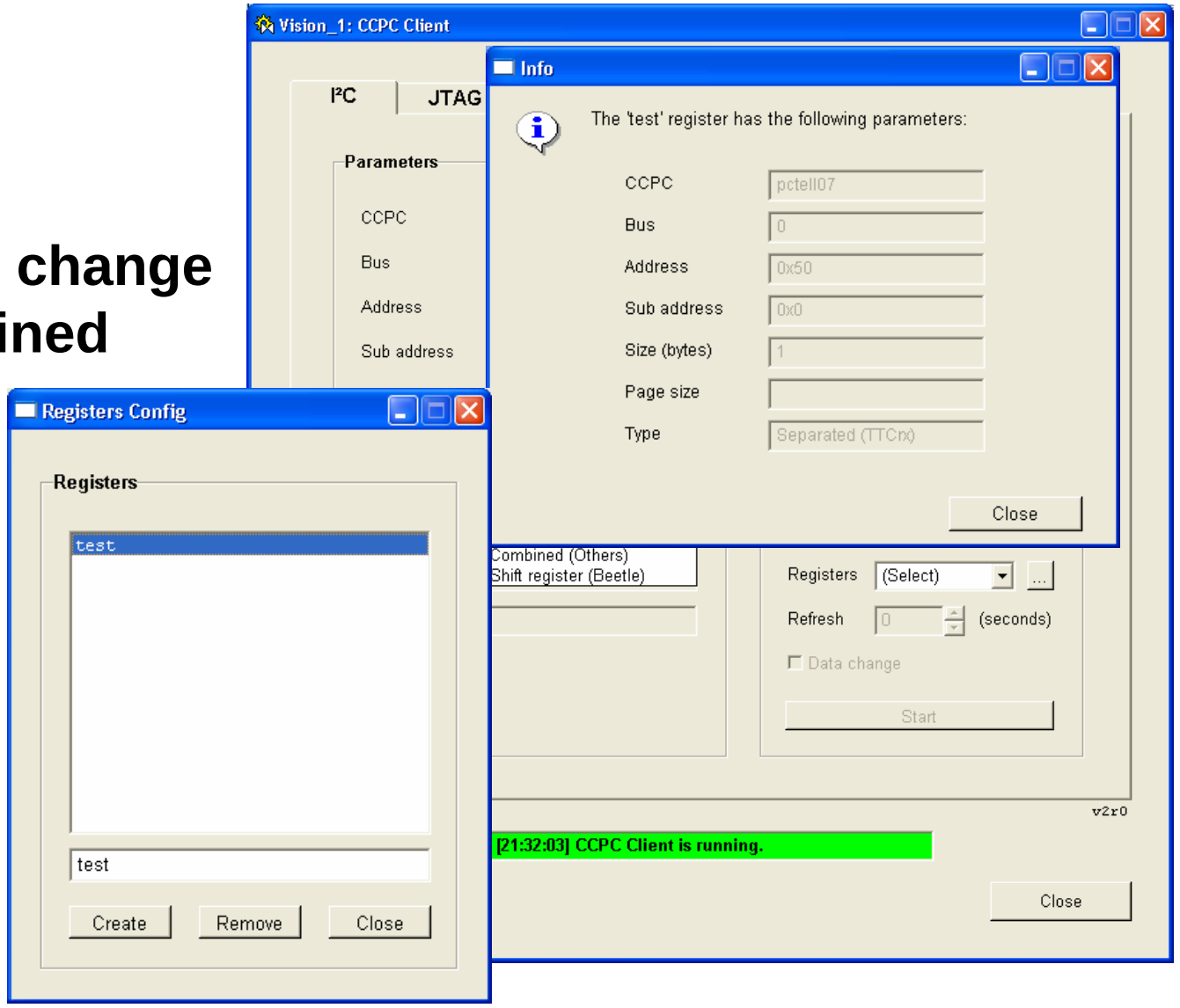
Use-case Tell1 board (IV):

I2C-scan:
scan the addresses of existing (sending acknowledge) devices.



Use-case Tell1 board (VI):

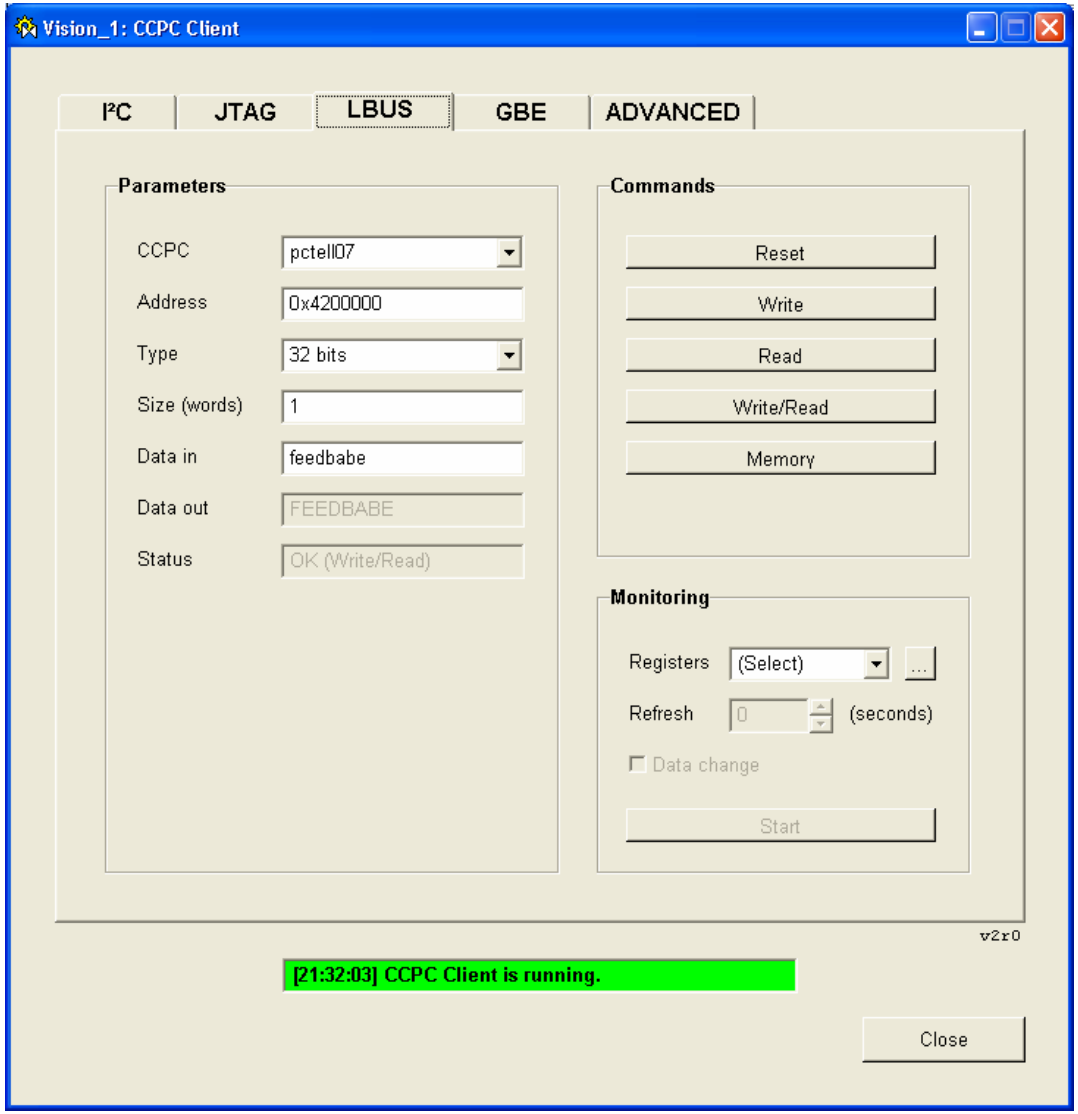
Monitoring:
select either on change and/or a predefined polling rate.



Use-case Tell1 board (VII):

LBus/GBE:
on the user-side they behave similar.

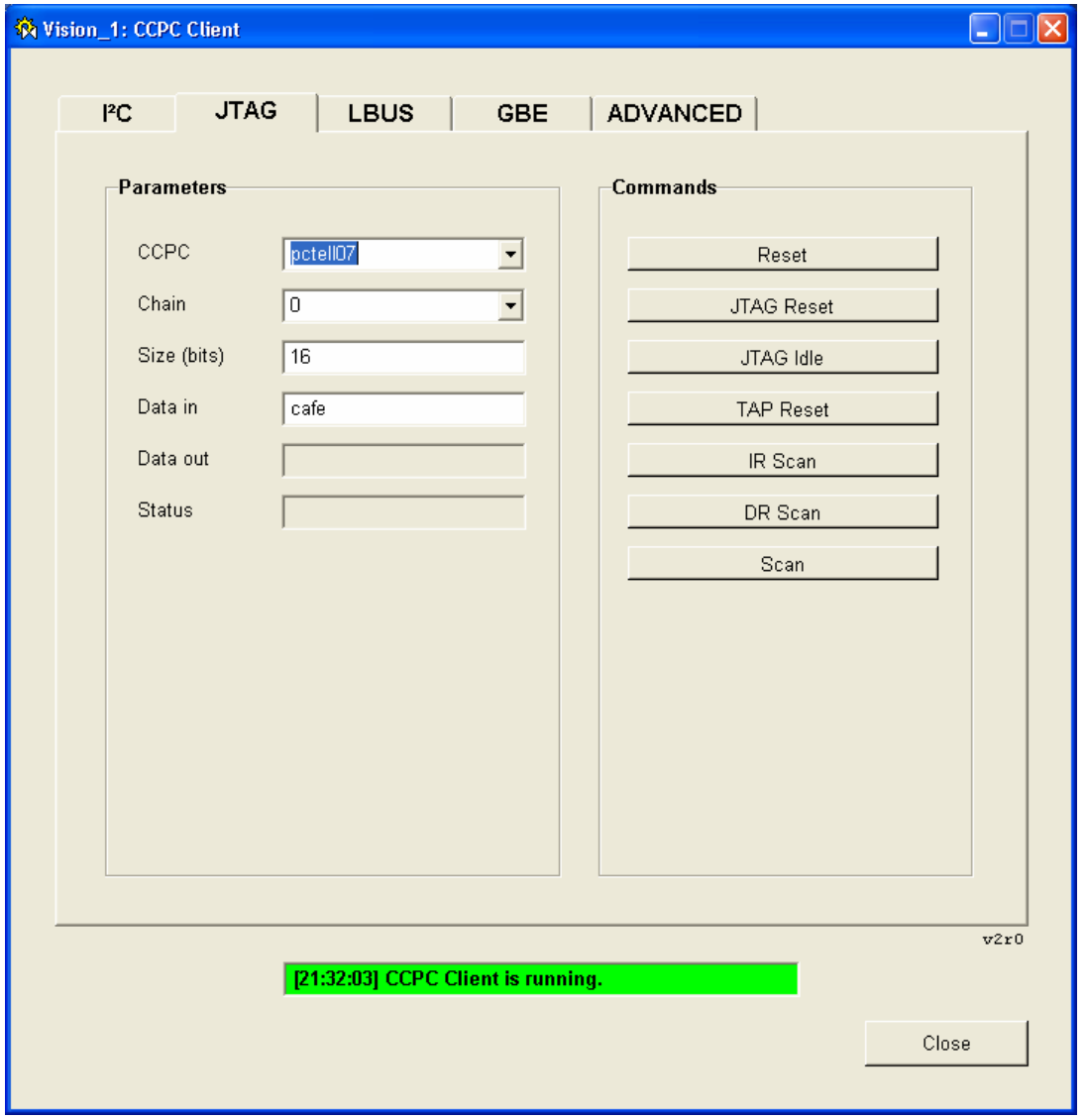
Internally however GBE is a 16 bit bus, where the low word is stored in a different memory block than the high word. (which is already treated at low level)



Use-case Tell1 board (VIII):

JTAG:
Basic JTAG Operations are supported.

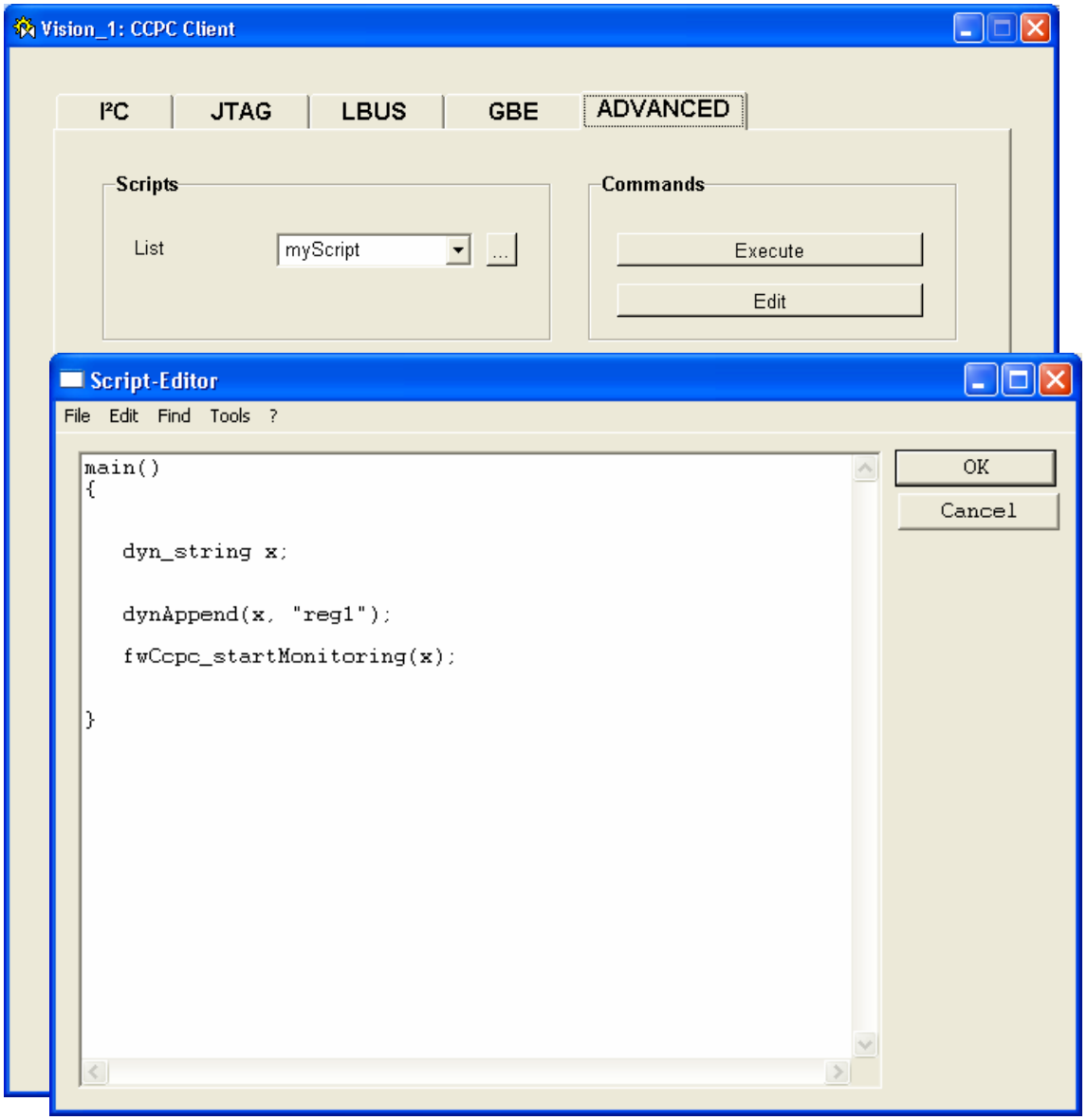
Integration into the generic FwHw-tool (for defining registers) possible.
Not really required for the Tell1 board.



Use-case Tell1 board (IX):

scripting:
little scripts can be run using the framework functions.

commandline tools:
can be launched from PVSS as well. (soon)

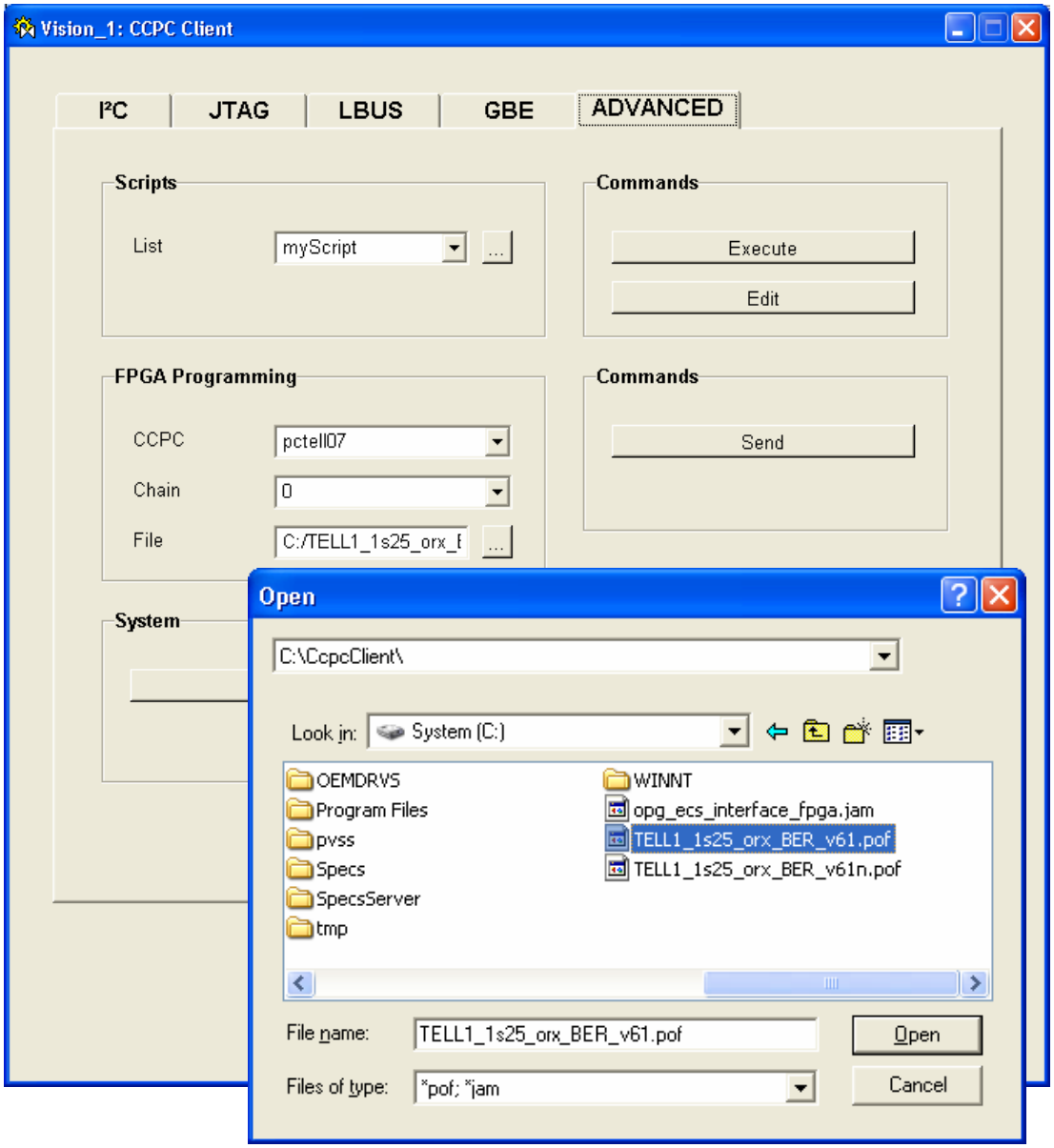


Use-case Tell1 board (X):

download of FPGA firmware:

supported files are .pof and .jbc (not .jam).

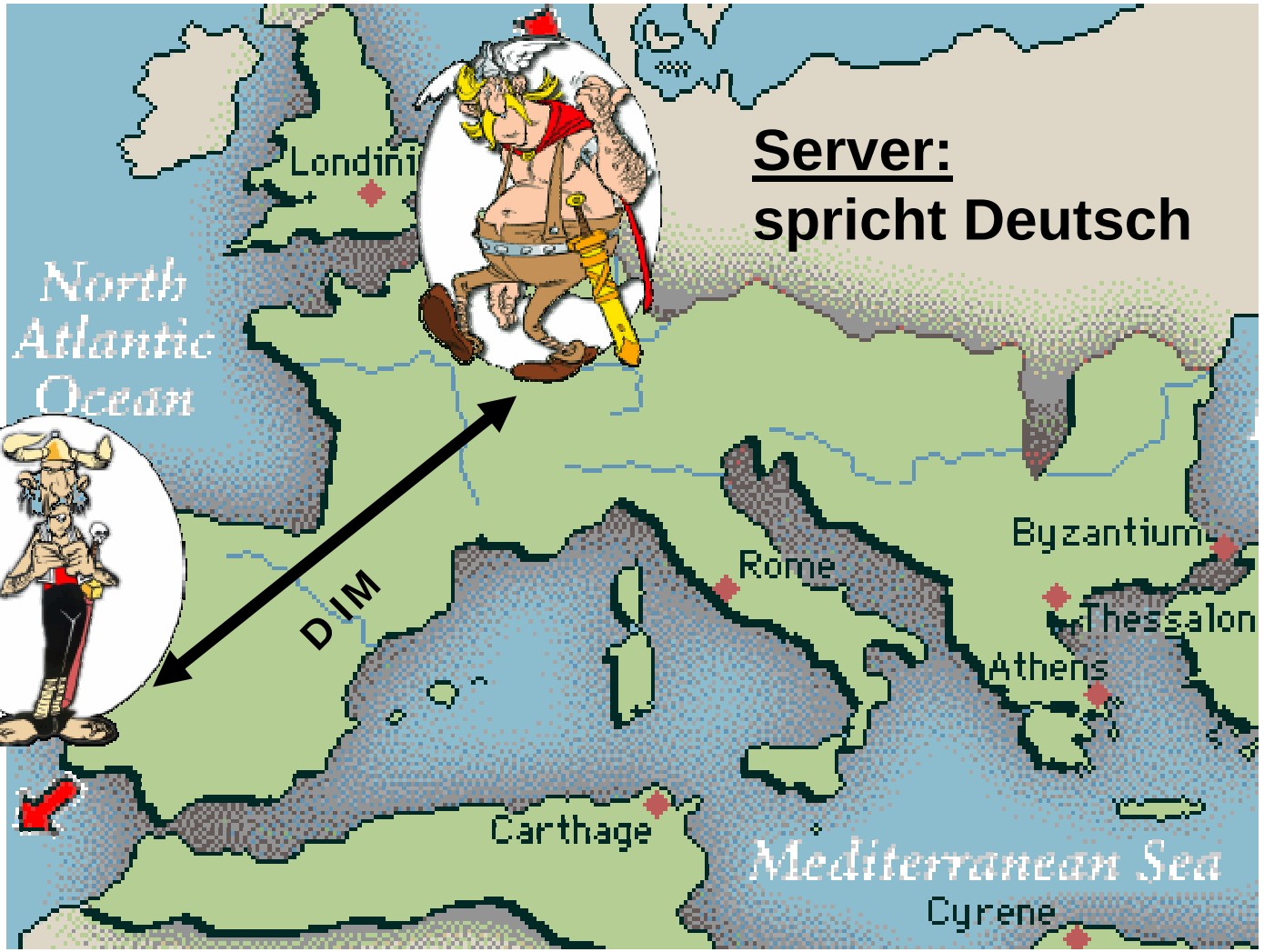
With jumper setting FPGAs can be programmed directly without EPC16!



Summary:

We have a running system which we want to distribute after some additional testing.(~2 weeks)
Comprises:
- I2C,LBus,GBE for register creation tool (FwHw)
- I2C,LBus,GBE,JTAG on test panels
- firmware download from PVSS
- scripting

Users should be able to use it while we take care of some (time consuming) details and improve it further (transparent). If you wish some special services on the server (which I have not thought of) please tell me!

Little Demo:

Or how to solve communication-problems with DIM