Plans for a smooth transition to CORBA in the Accelerator Control Software at LNL

S. Canella
L.N.L. - I.N.F.N.
Viale dell'Universita' 2, 35020 Legnaro (PD) - Italy
canella@lnl.infn.it

Abstract

LNL is an Italian National Laboratory [1] where 4 linear accelerators (Linac) for ion beams are in operation.
For the two major accelerators and the three main experimental halls large amounts of distributed software were developed in the past, to have remote control of magnets, diagnostics, RF resonators, bunchers.
A the beginning of 2000 the evolving technology forced the LNL control group to plan a transition to replace old C, X11, TCP-IP packages with new software standards. JAVA and CORBA [2], [3] technologies were chosen, both for the new applications and to update the old ones.
By this year a whole new application and two re-make have been planned an partially built, following these standards; they are now waiting for the test and debugging phase on field. They are all based on JAVA-Swing (for the GUI) but, in a first stage, the network layer still relies on TCP/IP sockets. As second stage, a prototype of a CORBA communication layer that may co-operate with the old style C servers will be prepared and tested.
In the LNL control software CORBA will be extensively applied after planning a precise but flexible architecture for the device/property model, together with the necessary interfaces towards the old services provided by the traditional well established software, as all old-style programs have to be maintained entirely running until the whole transition is smoothly completed.

1. LNL control systems

At LNL 4 Linacs are in operation delivering heavy ion beams of many different species and energies to targets in experiments for nuclear physics and other sciences. Three Linacs are old electrostatic machines, the most recent is an RF superconducting Linac (ALPI, built in the period 1990-95 and recently upgraded for better performances), used as booster of an electrostatic 15 MV XTU Tandem. An ECR ion source and a new superconducting injector (PIAVE) will start working next year, giving beam to ALPI in place of the Tandem accelerator.
The ALPI Linac now includes 71 independently phased superconducting (SC) quarter wave resonators (QWR, [4]); and may now reach an average equivalent voltage of 34.5 MV. The new injector following the ECR ion source consists of 2 superconducting radiofrequency quadrupoles and 8 more SC QWRs: when in operation it will reach an average equivalent voltage of 12 MV.
For the Tandem-ALPI couple (and their 3 experimental halls) large amounts of distributed customized software were in-house developed in the period 1990-95, to have remote control of RF resonators and bunchers (about 80), magnets (about 100) and diagnostics (about 30). This old control software was mainly written from scratch (the project started in 1990), using C and X11 for the GUI, C and TCP-IP for network communications (in client-server style, through sockets). These packages rely on UNIX workstations (which were DEC-MIPS Ultrix workstations in the beginning, then updated to Alpha-True64 and now a mix of Alpha-True64 and Sun-Solaris8 systems), VME IO systems (VxWorks) and a standard ethernet network at 10 Mbits/s. Due to its simple structure and to the experience gained writing the code, in about 10 years this software has been easily enlarged, updated, maintained, reused in small test sites, and, of course, used to run the LNL Tandem-ALPI accelerators and beam lines. Among the most remarkable past upgrading steps, two porting of the high level software (GUIs and network management) must be mentioned: from DEC-MIPS Ultrix to Alpha True64 in 1997 and from Alpha True64 to Sun Solaris a year later. Recently, magnets and RF control systems have also been ported to Linux Red Hat 7.2 PC. The VME IO systems server tasks were also easily ported from 68K to PPC processor boards.

2. The transition to CORBA technology (and to JAVA)

At the beginning of 2000 the evolving technology forced the LNL control group to plan a transition from C, X11 and TCP/IP frameworks to JAVA and CORBA both for the new applications and to update of the old ones. JAVA and CORBA have been chosen at the same time because CORBA is a very promising open and language-independent framework for communications in distributed systems and JAVA is an optimum language to write code for graphic interfaces and is also excellent for CORBA programs.

At the beginning the main reason to adopt JAVA-CORBA for control systems was the consequence of the announced project of a new high intensity proton Linac. Therefore, as a first step, the performances of CORBA for the control network services were measured in different conditions and with different platforms (in any case they showed to be sufficient to cover the requirements for remote operation [5]). After these encouraging results some prototypes of simple applications using the new technologies were then implemented and tested.
Besides, while designing and carrying out these applications other reasons for the JAVA-CORBA choices became clear:

- hopefully these choices will partially overcome the present problem of knowledge sharing of the control software for maintenance, especially if high level tools (such as JBuilder, Forte, ...) will be used to design and carry out new applications,
- JAVA and CORBA are both object-oriented, that is well suited for modern programming (and young programmers),
- JAVA GUIs and CORBA programs in JAVA generate bytecode, that is a code which may directly run on different platforms (even if at the price of being interpreted, so being less efficient than compiled code),
- CORBA is language-independent, so also C++ may be used, giving place to more efficient compiled-code distributed objects and services (available, for instance, also on a real time operating system like VxWorks)

Therefore in the short and middle period all new developments will rely on the JAVA-CORBA choices, while in the long period all (or at least most) GUI software will migrate to JAVA-Swing and most of the network layer will be converted to a JAVA-CORBA middleware.

3. Short term applications using CORBA middleware

In the current year three new applications have been planned: two of them are a re-make for the remote control of magnets and steerers of the ECR beam line, while the third is a new one, for the electrostatic lenses of PIAVE beam line and the high voltage power supply of the ECR platform. All the three applications are linked to remote controllers using RS232 serial lines. Magnet and steeres controllers for the ECR are connected to a local Linux industrial PC installed on the ECR platform. The electrostatic lenses power supply and the high voltage platform for the new injector are controlled by an HP 34970 general purpose controller connected to a LAT terminal server configured in an DEC-alpha workstation. The new GUIs were implemented in JAVA-Swing but their network layer, in a first step, still relies on TCP/IP sockets, as device servers have to be maintained compliant with the old architecture, as shown in Figure 1. The second step, in the next months, will be to build a prototype of CORBA communication layer that may co-operate with the old-style servers and, at the same time, make the new CORBA services available on the LNL network, as shown in Figure 2.
A prototype of the structure shown in Figure 2 was already implemented and successfully tested at the beginning of the current year for a step-motor controller connected to a Linux PC through an RS232 serial line using JAVA2 ORB as naming service.

4. Plans for the future

The actual stage for CORBA in the LNL control software is that of planning a precise but flexible architecture of the device/property model for a modern communication layer, together with the necessary interfaces towards the old services provided by the traditional software, as all old-style programs have to be maintained entirely running, for compatibility reasons, until the whole transition has been completed. Before starting a complete migration to the new communication standard a clear definition of the different CORBA objects create-use-delete policy cycle of life has to be defined, for each LNL family of control objects: magnets, steerers, faraday cups, beam profile monitors, RF resonators ... .
The purpose of the prototypes described in the previous section is only to show the feasibility of a transition to the new CORBA architecture, and that new JAVA-CORBA programs may work as well as (hopefully better) than the currently used packages.
However, to be effective from the long-term software management point of view this phase of object design is very important and worth of an effort that will take several months.

5. Conclusions

As CORBA use is still at an early stage at LNL, sharing experience and knowledge with people already working for long time on accelerators and telescope control systems will be of great help for our developments and any proposal of collaboration to set up standards and tools specific for the use of CORBA in these fields will be carefully considered.

6. Acknowledgments

We are grateful to the organizers of this CORBA controls workshop for the the opportunity they gave to people working in the areas of accelerators and telescope control systems to meet and discuss this specific subject.

References

[1] LNL home page - http://www.lnl.infn.it

[2] JAVA home page - http://java.sun.com

[3] OMG home page - http://www.omg.org

[4] A.M. Porcellato et al., "Upgrading of medium beta ALPI resonators by Nb sputtering", proc. of EPAC2002, Paris France

[5] S. Canella, G. Bassato, "Design of a Distributed Control System based on CORBA and JAVA for a new RIB facility at LNL", proc. of ICALEPCS2001, San Jose' California

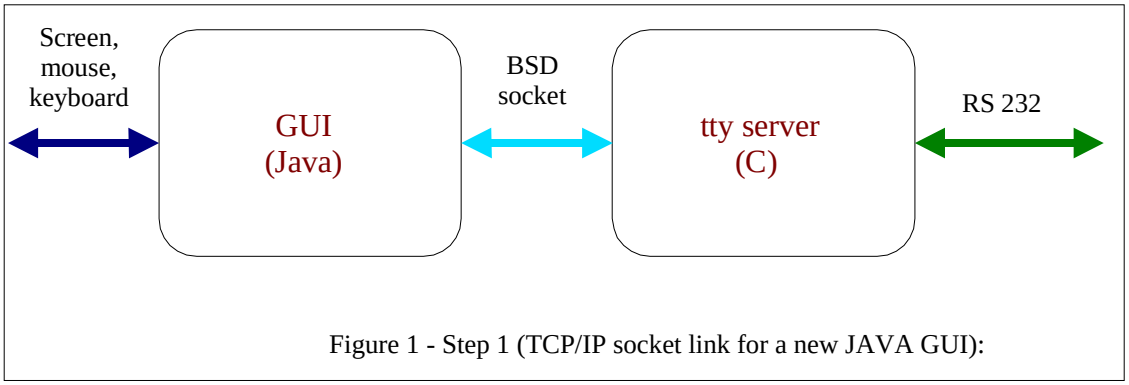

Figure 1 - Step 1 (TCP/IP socket link for a new JAVA GUI):

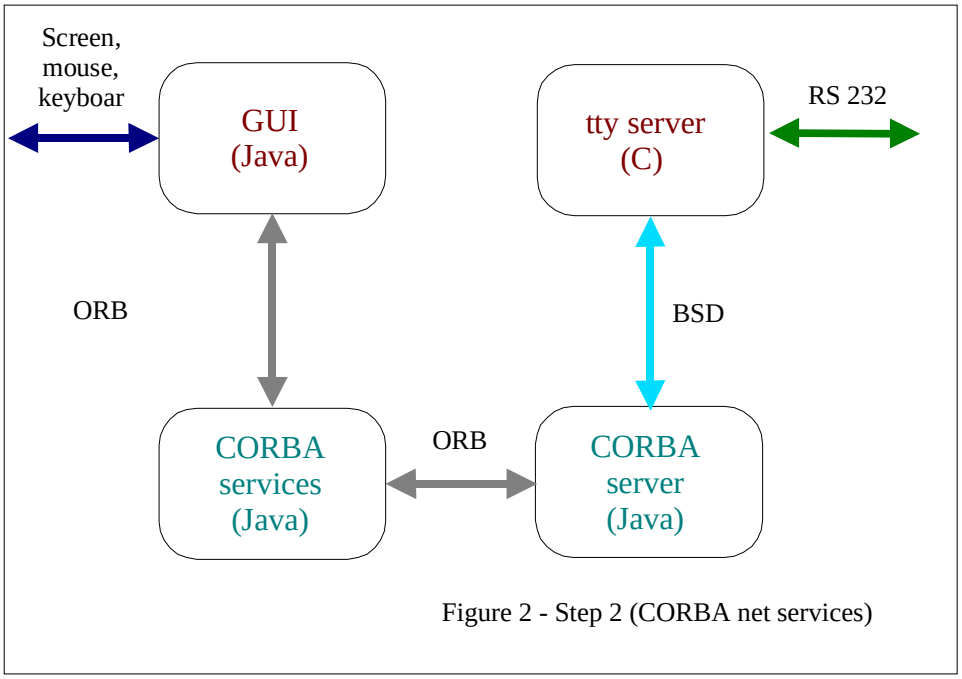

Figure 2 - Step 2 (CORBA net services)