

A LONG SUCCESSFUL JAPANESE-SLOVENIAN COLLABORATION IN CS DEVELOPMENT

G. Pajor*, I. Kriznar, Y. Oku, M. Plesko, I. Verstovsek, Cosylab, Slovenia
Toshikatsu Masuoka, Takeshi Nakamura, Koji Nakatani, NDS, Japan

Abstract

When Cosylab was founded 2001, Hitachi-Zosen (now NDS) was one of its first customers. Since then, we have successfully worked together on more than five control system software projects for various Japanese laboratories.

For these projects we both agreed on well defined interfaces and used some innovative ways of defining specifications and reporting bugs. All of this resulted in clearly specified tasks and helped us both avoid language barriers and misunderstandings. When NDS and Cosylab parts were put together they fitted nicely and no on-site installation of Cosylab products was ever required.

It is important that all sides comply to the interface specifications, so the extra work for fitting pieces together is minimal and the possible problems can be addressed to the group responsible for the particular part.

We have used IDL (Interface Definition Language) and simulation servers for defining and testing the programming parts. For specifications and bug reports we used power point presentations and edited screenshots. All inter-company communication has been done via email.

Our experience show that this kind of collaboration is very cost-efficient for both sides, but also requires both sides to be responsible partners.

In this article we will discuss the collaboration between NDS and Cosylab as an example of successful collaboration between Japanese and European company.

OVERVIEW

From 2001 to present day, NDS [1] and Cosylab [2] collaborated on projects for UVSOR [3], SuperSOR [4], RIBF [5], JAERI [6] FEL, JASRI [7]. NDS developed the control and data acquisition processes on the server, while Cosylab developed the GUI and all the logic related to it.

The software Cosylab developed for NDS is based on Abeans R2 control system libraries. The lower level of of Abeans R2 application, called *plug*, communicates with server through CORBA. The server contains objects that represent the controlled devices. The higher level, graphical user interface, enables user to monitor and control the objects on server.

COMMUNICATION AND SPECIFICATIONS

The language barriers could be a problem, but in this collaboration, they were overcome with a variety of approaches and a lot of goodwill among the partners.

All the communication between NDS and Cosylab was done via e-mail. Cosylab provided a mailing list, to which all involved developers from Cosylab and the responsible

contacts from NDS were subscribed. The language of communication was mostly English, but in some cases, to avoid misunderstandings, Cosylab provided the translation to Japanese.

A Picture is Worth a Thousand Words

The specifications for the graphical user interface (GUI) were supplied by NDS in powerpoint presentation format. As it might not be a standard format for specifications, the powerpoint is well known, wide spread and provides easy to use composition of graphical elements, diagrams, text, etc. For these reasons it proved to be a wise and practical choice.

The desired appearance of the GUI was composed of existing screenshots' parts rearranged and pasted together. Any missing parts were drawn on top of the composed image.

The functionality of the GUI was indicated by connecting the GUI components to the relevant part of IDL. The comments were added where needed to avoid confusion or to explain additional functionality (see Fig.1).

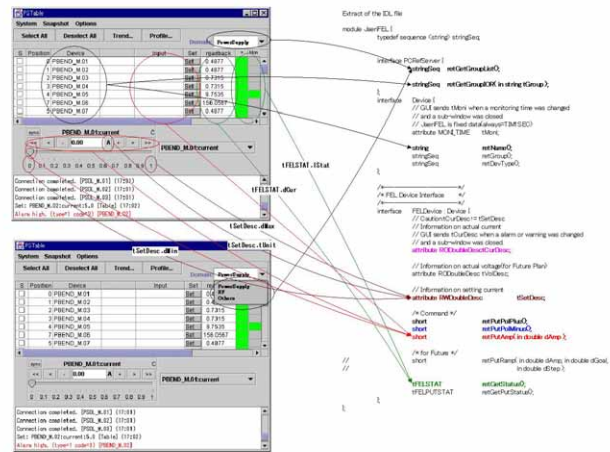


Figure 1: Example of GUI specifications.

This kind of GUI specifications minimized the time needed by NDS to prepare them as well as the time needed by Cosylab to understand them. Even more important, it eliminated almost all of possible misunderstandings; the pictures and diagrams were more verbose and less language-dependant than ordinary text would be.

Interface Towards Server

The Abeans - server communication interface was defined in IDL (interface definition language). Cosylab suggested device/property model for IDL to enhance the compatibility with existing Abeans libraries. Each project

* gasper.pajor@cosylab.com

has its own specifics regarding IDL, so there were always some modifications of the Abeans plug necessary. Both sides tried to reach an agreement on IDL so that these modifications would be minimal.

TESTING AND DELIVERING THE SOFTWARE

Simulation server

Testing plays an important role of software development. One of best ways for testing the functionality and behaviour of control system software is to run it where it would ultimately be installed, together with all the complementary (e.g. lower-level) software and real hardware. However, because of development on remote location, this could not be achieved in our case.

The closest approximation was testing with simulation server that would simulate the lower layers of control system. NDS provided such server for each project so application could be tested against it during development.

NDS also provided sources for the server, so the simulation parameters could be changed.

It turned out that this kind of approach was very successful and served the testing purpose well.

Delivering an Installer

When the development of the control system software is finished, usually the on-site installation is required and some modifications are needed to fully integrate the software with the rest of control system. Due to Cosylab's development in Europe, the on-site installation in Japan would mean considerable expenses.

Among all other problems, this one was also eliminated by well defined interfaces, specifications and the simulation servers. For projects Cosylab and NDS collaborated on, the on-site installation was not necessary.

customer pages (see Fig.2) in a form of Windows installer. NDS had only to download and run the installer and the applications were ready to run.

The code that was developed specifically for NDS was also made available to them.

Bugs and Feature Requests

As much as we try to avoid it, there always is a possibility for a software to behave in a way different than intended. For reporting bugs, feature requests and other issues, NDS utilized the same approach as for specifications – the powerpoint files. They carried the graphical and textual explanation of conditions and manifestation of the problems in the case of bugs, or an updated variant of specifications in case of feature requests (see Fig.3).

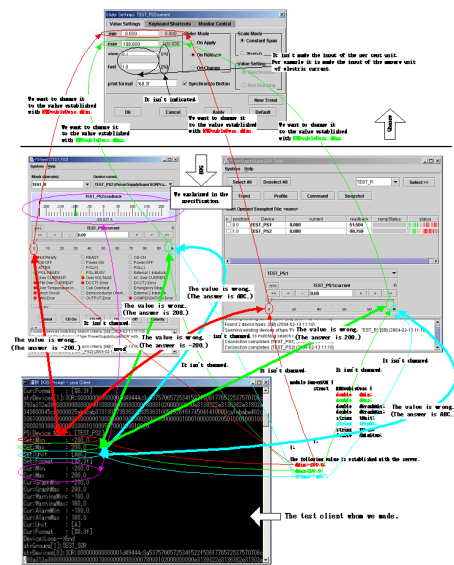


Figure 3: Bug Report Along with Some Feature Requests.

Cosylab provided jar files with fixes and when the solution was confirmed, a new version of installer was made available on web.

INTERNATIONALIZATION

For RIBF project, NDS requested the localization of CS applications. Cosylab faced a new challenge, as the Abeans R2 graphical libraries were not designed for multi-language support and such support had to be retrofitted into existing code.

For a program to be localized, it first needs to be internationalized, meaning that all locale-sensitive data (GUI text in our case) have to be extracted from the source code. Configuration and translations for each supported language have to be set up and, of course, the mean of language selection has to be implemented.

We made use of Java built-in support for internationalization.

All relevant English text was extracted to a separate file. To assure backwards compatibility and avoid problems with missing text, the English version was still used as default and was included in the sources of the applications.

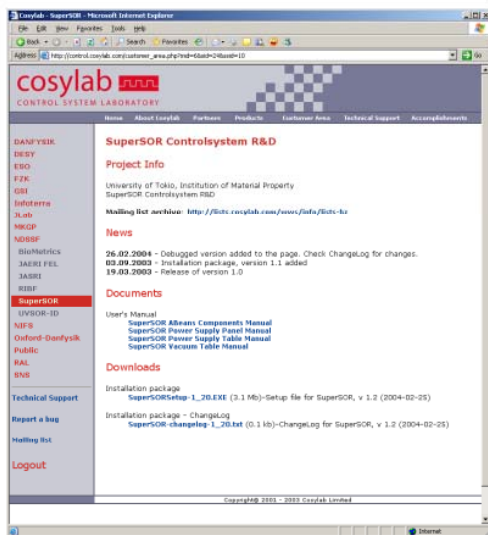


Figure 2: Cosylab Customer Page for SuperSOR project.

The Abeans R2 libraries, the plug to NDS CORBA server and the graphical applications were packed in standard jar files and were published on the Cosylab

Japanese translation of the English text was added as a separate text file, so it could be easily edited and corrected.

Some of the GUI text, however, was not stored on client machine, but was received from the server instead. This was the case with generic commands' names and alike.



Figure 4: Partially Translated Application and Language Selection Dialog.

We implemented a simple language selector which prompted user for language selection. Based on the user's choice, the language file was loaded as well as the appropriate methods were called to get either Japanese or English generic commands' names from server.

CONCLUSION

NDS and Cosylab have proven that control system software can be successfully developed on two hemispheres, despite the language barriers and cultural differences. Although unorthodox, many of the methods developed here can be used also for large collaborations such as the GAN. We therefore see our contribution in this paper not so much on technical excellence, but as a sociologically successful experiment, which is perhaps even more difficult to achieve for us technical people. It would not have been possible without the honesty and forthcoming of the key people on both sides.

Of course off-site development would become somewhat more difficult with larger projects, but could, with imaginative approach and good specifications, still be done.

When software from both sides was put together, it fit as expected, without the need of on-site installation. We consider this an accomplishment worth mentioning, as this is rarely the case in control system development.

The years of our collaboration also prove that Cosylab can meet the demands from a Japanese company in a tough and competitive environment. We hope that our fruitful collaboration with NDS and possibly other Japanese companies will continue. It has been a challenge to start, but a pleasure to succeed.

ACKNOWLEDGEMENTS

Cosylab would like to most warmly thank Takeshi Nakamura from NDS for arranging and managing this long and fruitful collaboration and his support regarding this article. Mr. Nakamura also arranged the agreements for usage of names of Japanese laboratories in this article. Cosylab sincerely thanks Toshiya Tanabe for establishing the first contact between NDS and Cosylab and helping to maintain it.

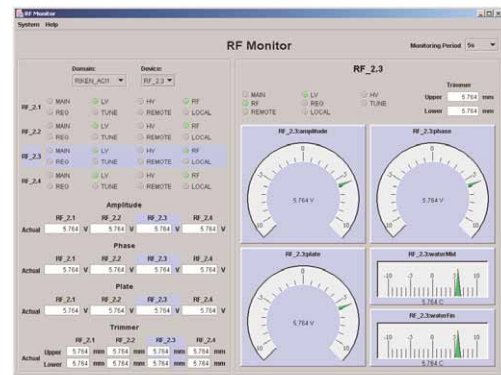
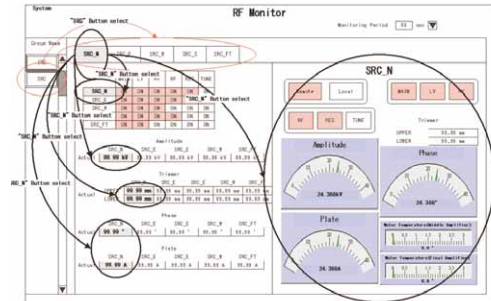


Figure 5: Specifications from NDS (top), Actual Implementation by Cosylab (bottom).

REFERENCES

- [1] NDS, <http://www.ndssf.co.jp/>
- [2] Cosylab, <http://www.cosylab.com>
- [3] UVSOR (Ultraviolet Synchrotron Orbital Radiation Facility), <http://www.uvsor.ims.ac.jp/>
- [4] SuperSOR
<http://www.issp.u-tokyo.ac.jp/labs/sor/project/>
- [5] RIBF (RIKEN RI Beam Factory) ,
<http://ribfweb1.riken.go.jp/>
- [6] JAERI (Japanese Atomic Energy Research Institute),
<http://www.jaeri.go.jp/>
- [7] JASRI (Japan Synchrotron Radiation Research Institute), <http://www.spring8.or.jp/>