

Slovenian President's First Visit to Japan

BY: M. PERTOVT (COSYLAB)

The President of the Republic of Slovenia, Mr Borut Pahor visited Japan between 6 and 9 March 2013. The President was accompanied by a 33-member delegation made up of representatives from Slovenian businesses, especially in the sector of high technology.

The visit was organized by the Ministry of Foreign Affairs in cooperation with the Embassy of Japan, the Slovenian Public Agency for Entrepreneurship, Innovation, Development, Investment and Tourism (SPIRIT Slovenia) and the Office of the President of the Republic. Many business meetings, presentations and events, such as the Slovenian-Japanese business investment forum and "business to business" (B2B) meetings were arranged by SPIRIT, the partner agency in Japan, JETRO, and the Japanese New Energy and Industrial Technology Development Organization (NEDO) in Tokyo and Yokohama.

NEDO, hosted the Japan-Slovenia Technology Cooperation Seminar between relevant Japanese and Slovenian businesses to share information about various NEDO initiatives in fields such as smart communities.

SPIRIT Slovenia and NEDO have signed an agreement on cooperation in the field of



President Borut Pahor, with Cosylab Japan members Takashi Nakamoto (left) and Dr. Shin-ichi Kurokawa (right).

advanced industrial technologies. NEDO is currently funding a study by PWC and Hitachi is working on opportunities in Slovenia to set up a pilot project around *smart communities* with the involvement of Slovenian companies.

The main objective of cooperation (and the focus of the most of the discussions) is setting up and developing bi-directional flow of technologies between Japan and Slovenia in the cross-sectional fields of new energy technologies and process control technologies, balancing the created added values, collaboration of companies in both domestic markets and foreign markets, and increasing the volume of operations for both sides.

The business activities of partners will cover carrying out joint R&D, pilot and

demonstration projects for new solutions into the market; carrying out engineering projects; and implementing/selling solutions, systems and products.

Cosylab's interest in participating comes in on users and distributors of control systems for large scientific installations, like particle accelerators and nuclear fusion, distributed control systems, system integration of large IT installations and PV inverters, specifically micro-inverters, which Cosylab develops and manufactures.

Early in 2013, six companies set up a European Economic Interest Grouping (EEIG) Japanese-European Technology center for new technologies and energy management (JETNET, EEIG).

The founders of EEIG in Slovenia are: INEA d.o.o., Cosylab d.d., Centre ARI, Robotina d.o.o, KIV d.d. and in Italy: EUROSERVIS - S.R.L.

EEIG's main goal is increasing or developing the business activities of its members in cross-sectional fields and to improve or augment the results of these activities. Representatives of these companies also formed part of the delegation that visited Japan.



NEWS SNIPPET: ALMA INAUGURATION!

The official inauguration ceremony for the Atacama Large Millimeter/submillimeter Array (ALMA) [1], the largest astronomical project in existence, was held on 13 March 2013, in Chile. The inauguration marks the completion

of the major systems of the telescope array and formally indicates the transition to a full observatory. ALMA is an international partnership between Europe, North America, and East Asia in cooperation with the Republic of Chile.

Cosylab has worked on the ALMA control system since ALMA infrastructure develop-

ment started in 2000, technically since before there was a "Cosylab").

Slovenian television featured a segment about Cosylab's involvement in the ALMA project during the evening news on 14 March 2013.

[1] <http://www.almaobservatory.org/en/home>

Risks, Requirements and Agile Development

BY: K. MEYER (COSYLAB)

So you have this amazing control system problem and you'd like Cosylab to solve it or implement it for you. Typically this means that you are one of two kinds of customer: Either you know exactly what you want and you have all your requirements fully identified or you don't.

If you do, then according to some ISO/IEC standards, for example the ISO/IEC 15288 standard on "Systems and Software Engineering – System life cycle processes", you would have identified all your stakeholders, had a few rounds of stakeholder requirements meetings, and have fully documented their expectations and requirements – and confirmed that the documented requirements are representative of what the users actually said. If all went according to plan, you would have a list of all your requirements and who wanted what (and why). Those last bits are important so that when it comes round to turning requirements into specifications, and some requirements are unable to be accommodated, you know who wanted them. Then you can go back to them and give them the bad news.

Of course, according to the same standard, you should also have identified (amongst other things):

- constraints on any solutions that are the consequence of existing agreements and decisions,
- interactions between the users and the required system, and
- health, safety & security requirements!

Depending on your technical capabilities at this point you can either turn the user requirements into a system specification yourself (remembering to preserve the mapping between the requirements, that is who-wanted-what, in the resulting specification), or you can contact Cosylab and we can work with you to produce the specifications. With a complete and well scoped set of specifications Cosylab can provide you with a quote and delivery schedule that is accurate and fair to all parties.

If, on the other hand, you have only a vague

idea of what you want and can't produce a comprehensive requirements document (never mind a detailed specification) then things are a bit trickier. When your requirements are ill-defined the Cosylab team needs to do a lot of guesswork (guided by experience, but there is still guesswork involved) to

deliver a growing software project at regular (short) intervals. Every few weeks or so you get the next iteration of a growing solution. This way you get to quickly see the solution in action (you get to use it) and

you can guide development. If it becomes clear early on that your problem can't be solved, then you can terminate the development early and reconsider your options.

What are the risks?

In this kind of project, you are being honest up front that you don't know the project scope, and you don't know how much the solution will cost. You can't give your buying office a definite project cost and you can't tell management how long it will take. The budget you unlock buys you a fixed development effort and the complexity of the problem determines whether the development effort you've bought is able to deliver an acceptable solution.

How do you manage this risk and uncertainty?

Cosylab project architects would need to spend some time with

you to gauge some of the project specifications, and propose a sequence of deliverables that provide you with more and more functionality, while ensuring that each deliverable is a functional improvement. This means that from the first deliverable you receive a product that works, but with limited (or very specific) functionality. Then, iteration by iteration, this initial product is extended, requirement by requirement, until either you have the control system you want, or your budget is exhausted (and when you unlock more budget, you can continue the development). However, for this approach to succeed, the project must depend on only existing hardware and you must have someone available to receive and use the solution each delivery cycle. This person (or team) must be available to use the solution and provide feedback to the Cosylab development team.



In the classic development cycle, all the requirements are determined during an investigation phase and are used to design, develop, test and deploy the product. In agile development, repeated cycles of partial requirements gathering, design, development and testing are used to repeatedly release a product with growing capabilities.

try and estimate the specification and the effort required to deliver the final product. This can lead to unfairness and disappointment. Unfairness, because if the actual work required turns out to be a lot more than estimated, Cosylab absorbs the extra work. Disappointment results when the required effort turns out to be significantly more than expected and Cosylab needs to negotiate an extension to the contract!

What's the solution?

Ideally, try and ensure that you know precisely what it is that you want and draw up decent requirements. Cosylab can help out: make the first task order a requirements analysis project.

Are there any alternatives?

Yes, actually. Cosylab also offers so called "time and material" contracts, so why not try an agile project: For a fixed rate, Cosylab can



(CONTINUED ON PAGE 4)

EPICS Data Acquisition Device Support

BY: V. ISAEV (COSYLAB) AND K. ŽAGAR (COBIK*)



In EPICS, the functionality of devices integrated into the control system is exposed through records. EPICS pre-defines a set of record types (ai, ao, mbbi, etc) and these are sufficient for simple devices that don't need any provisions for their state management and configuration. However, to model devices that are more complex, either a new record type needs to be introduced (e.g., `steppermotor`), or a device can be exposed via multiple records. While in the latter approach there is no *a priori* standard by which to structure and name records pertaining to a device (a "PV interface"), such a standardization would be beneficial.

Several standards to this effect do exist, for example `areaDetector` for image acquisition devices [1] and `Generic Transient Recorder` for externally triggered data acquisition devices [2]. To this collection, we have added the *Nominal Device Model (NDM)* which is designed to standardize the EPICS interface for analog and digital input and output devices. It can also be used to support image acquisition devices (cameras) and timing devices (masters, receivers, delay generators, event generators).

NDM BENEFITS

Facilitate re-use, e.g., a state machine or a user interface designed for one device can be used with another device.

Reduce documentation. The interface needs to be described only once and applies to all its implementations.

Simplify requirements and project management. Knowing a superset of all functions that an interface provides, it is possible to unambiguously and efficiently specify what functions are to be implemented and how.

Reduce device integration effort. Standard might ship with a support library that facilitates implementation of device-specific device drivers.

Reuse the test suite. Automated tests written against the interface can be used to test a newly implemented device driver for compliance.

each image (either raw or processed) can be modeled as a channel. Similarly, for generators of timed events, each trigger line can also be considered as a digital output channel.

Apart from the structural model (class hierarchy), NDM also defines a behavioral model of devices and channels. The state machine allows differentiation between normal (on/off/idle/busy) and fault states (temporary/permanent malfunction).

For advanced configuration of the device, a messaging mechanism is provided so a client can send a message to the device or channel. Messages are passed as strings through character waveform records and convey commands and their responses. Messages are used e.g., to request a state transition, to upload firmware, etc.

NDM is written in C++ and is based on the `asynPortDriver` [3], thereby providing an efficient integration into an EPICS Input/Output Controller (IOC). The class view of the NDM is shown in Figure 1. Each device consists of multiple *channel groups*, which in turn consists of multiple *channels*.

Channel groups are used to bundle channels that are not independent of each other. E.g., on some devices, the sampling frequency cannot be set on the level of an individual channel. We have found such device/channel abstraction to be rather powerful, as it is applicable to all kinds of devices that NDM is designed to support. For cameras, for example,

Common functionalities supported by the NDM are:

- Signal acquisition as single value or waveform.
- Continuous and triggered data acquisition.
- Signal filtering.
- Fourier transforms of signals.
- Conversion of units using piecewise cubic spline curves, configurable at run time.
- Firmware upload.
- State management at the device and channel level.

For more information about NDM, including obtaining access to the code, please email klemen.zagar@cosylab.com.

References

[1] M. Rivers et al.: `areaDetector`: <http://cars.uchicago.edu/software/epics/areaDetectorDoc.html>

[2] M. Kraimer, E. Norum et al: `GTR – Generic Transient Recorder`: <http://www.aps.anl.gov/epics/modules/analog/gtr/index.html>

[3] M. Rivers: `asynDriver: Asynchronous Driver Support`. <http://www.aps.anl.gov/epics/modules/soft/asyn>

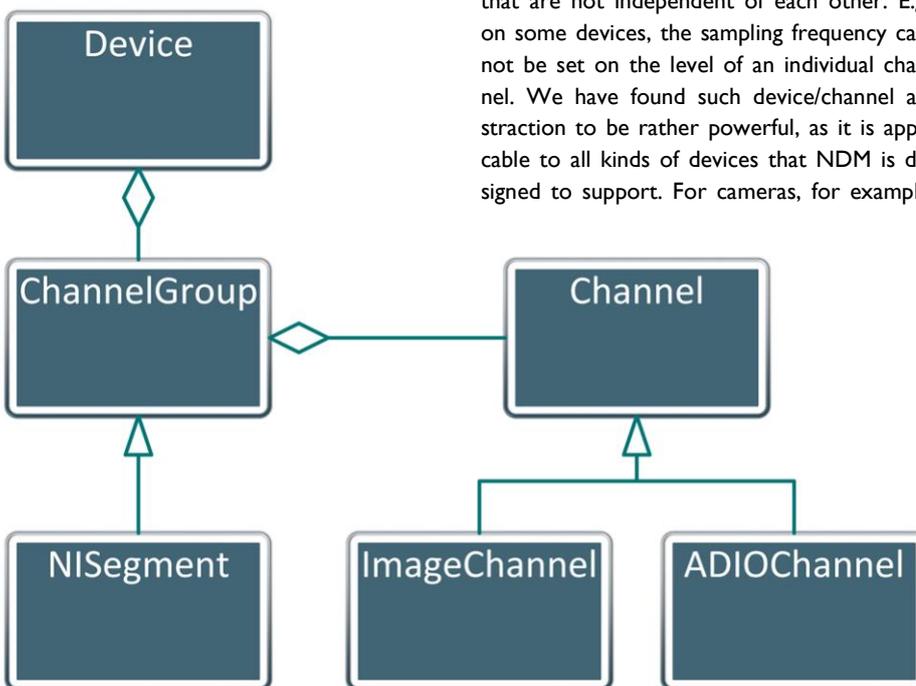


Figure 1: Class diagram of the nominal device model.

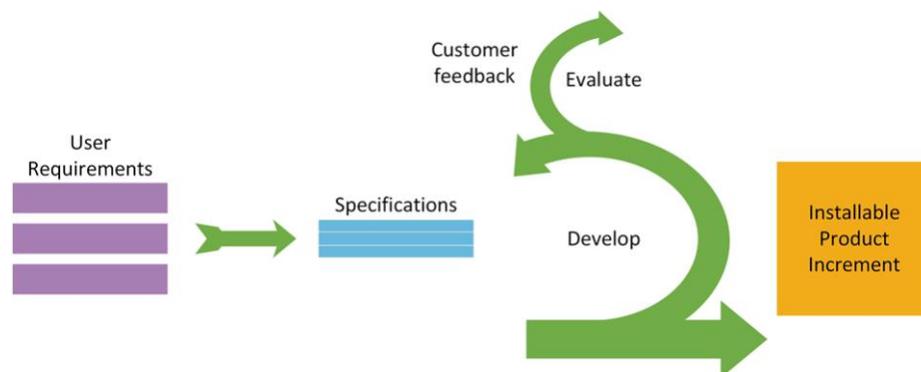
*Centre of Excellence for Biosensors, Instrumentation and Process Control (COBIK), Velika pot 22, SI-5250 Solkan, Slovenia COBIK is financed by the European Union, European Regional Development Fund and Republic of Slovenia, Ministry of Higher Education, Science and Technology."

Risks, Requirements and Agile Development

(CONTINUED FROM PAGE 2)

Traditionally, the development team stops the development effort for a few days to allow you to evaluate the product and provide feedback. This pause helps to prevent the product from drifting too far from what you are discovering you want. Remember – this method assumes that you don't have a well-defined set of requirements and Cosylab is doing two things: Firstly, we are helping you uncover your specifications, and secondly, we are delivering a solution that grows to solve your problem.

If none of these methods are a good fit for you, then you can always combine the two main ideas: commission Cosylab to deliver a smaller functional pilot, with a fixed cost / fixed duration contract (with the best estimate that Cosylab can provide, based on your available requirements) to formally develop a par-



In agile development, initial requirements are used to create a simple specification which in turn is used to rapidly develop an installable product. During the development, customers evaluate the product and provide feedback. The entire process is repeated so that the product grows and gains more capabilities with each additional development cycle.

tial pilot solution. Some of this pilot would be used to explore your greater requirements. By the end of the pilot, Cosylab would have a

much better understanding of your requirements and you will get a much more accurate quote and delivery schedule.

NEWS SNIPPET: COSYLAB JOINT VENTURE

Cosylab d.d. has formed a 50-50 joint venture with Letrika d.d. from Nova Gorica, Slovenia. The new company, Letrika Sol d.o.o. will draw on expertise from both parent companies to focus on all aspects of the development and marketing of micro-inverters [1] and their

integration into smart electric grids and e-services.

Letrika is well known for their motors and power supplies to the auto-motive industry. Letrika, interested in expanding to other markets, is developing the power electronics for a micro-inverter, and initially asked Cosylab to develop the communication part and the serv-

er side software, which will connect and manage many micro-inverters together in a solar power plant. However, during discussions it emerged that Cosylab could contribute much more to the project and it was agreed to form the joint venture company.

[1] http://en.wikipedia.org/wiki/Solar_micro-inverter

Invitation to Submit Content for Control Sheet

Control Sheet readers are invited to submit content for upcoming newsletters and get a Cosylab T-shirt for your trouble. You can submit technical control system articles, news items and pictures of you and your colleagues in your favourite Cosylab T-shirt. Send your contributions to controlsheet@cosylab.com.

Cosylab d.d.

Teslova ulica 30
SI -1000 Ljubljana
Slovenia

Phone: +38614776776

Email: controlsheet@cosylab.com

Web: www.cosylab.com

Visit us on: http://www.cosylab.com/resources/our_newsletter_control_sheet/