UNICOS EVOLUTION: CPC VERSION 6

E. Blanco Vinuela, J.M. Beckers, B. Bradu, Ph. Durand, B. Fernandez Adiego, S. Izquierdo Rosas, A. Merezhin, J. Ortola Vidal, J. Rochez, D. Willeman
CERN, Geneva, Switzerland

Abstract

The UNICOS (UNified Industrial COntrol System) framework was created back in 1998. Since then a noticeable number of applications in different domains have used this framework. Furthermore UNICOS has been formalized and its supervision layer has been reused in other kinds of applications (e.g. monitoring or supervisory tasks) where a control layer is not necessarily UNICOS oriented. The process control package has been reformulated as the UNICOS CPC package (Continuous Process Control) and a reengineering process has been followed. The drive behind these noticeable changes was (1) being able to upgrade to the new more performing IT technologies in the automatic code generation, (2) being flexible enough to create new additional device types to cope with other needs (e.g. Vacuum or Cooling and Ventilation applications) without major impact on the framework or the PLC code baselines and (3) enhance the framework with new functionalities (e.g. recipes). This publication addresses the motivation, changes, new functionalities and results obtained. It introduces in an overall view the technologies used and changes followed, emphasizing what has been gained for the developer and the final user. Finally some of the new domains where UNICOS CPC has been used will be illustrated.

INTRODUCTION

UNICOS (UNified Industrial Control System) was born more than a decade ago for the development of the LHC cryogenics control system. Since the first application deployed in mid 2001 the enrichment of functionalities and upgrades followed one after the other [1].

Initially, the UNICOS aim was the automation of an industrial process covering the layers of supervision and control. SCADA (Supervisory Control and Data Acquisition) and PLCs (Programmable Logic Controllers) were respectively the components used in those layers. For the first implementation, PCVue32® as SCADA and Schneider PLCs for the PLC layer were the choices. The addition of the Siemens S7 PLCs and the replacement of the SCADA by PVSS (nowadays called WinCC Open Architecture) was the natural evolution as they became recommended CERN components for an industrial control system.

The UNICOS conceptualization in a framework created not only a well defined set of objects but also: (a) an approach to model a process in units (IEC-61512), (b) a generic environment to develop the application specific control logic and (c) a unified way of operating an industrial plant [2].

During the last years the UNICOS framework has been not only successfully applied to the LHC cryogenics but also to many other applications (e.g. gas flow, collimator temperatures, equipment cooling, etc.) and lately to other different domains like ventilation and vacuum installations [3].

The maturity of the framework has induced an evolution towards even a more reusable and open model where the components provided by the framework are not only restricted to PLCs in the control layer [4]. In this way, UNICOS has been also applied to supervision and monitoring applications as the QPS (LHC Quench protection systems), SURVEY (alignment of the focusing LHC low beta magnets) or the PIC (Powering Interlocks controllers) among others.

The process control package has been reformulated as the UNICOS-CPC package (Continuous Process Control) and a sound reengineering process has been followed to finally create the UNICOS-CPC v6.

THE REENGINEERING APPROACH

After more than 10 years of usage of the framework in different domains, process and automation engineers together with plant operators have naturally identified several potential improvements and new requests.

The LHC detectors Gas Control project created additional objects to cope with missing framework functionalities (i.e. parameterization, simple status and analog alarms) which were not initially included in the basic package.

One of the framework features is the use of tools to automatically generate the code; those tools (based on Microsoft Access) although very valuable were unnecessarily rigid for the creation of new object types and rather inefficient when dealing with large projects.

The application dependent control logic is composed of a set of templates that combine together PLC code and a meta-language to make use of advanced features during the code generation (e.g. code reusability, user parameterization, etc.). Users regularly reported difficulties in understanding those logic templates.

All these facts were the driving force behind the vigorous reengineering process of the UNICOS-CPC package.

NEW FEATURES

Some of the package enhancements are: (1) new improved functionalities in almost all its objects as well as new objects, (2) extended flexibility to ease the creation of new objects, (3) the new tooling suite for the automatic generation of the applications, (4) improvement readability and usage of logic templates, (5) local operation enabled, (6) the change of the Siemens PLC application architecture, (7) dynamic recipes and, finally (8) HMI improvements.

Basic objects improvements

There has been a rationalisation of the existing CPC objects trying to keep their generic characteristic yet. This task has ended by the removal of redundant objects (e.g. Digital Input Calculated is integrated in the Digital Input), deletion of non used objects (e.g. Computed) and creation of new objects (e.g. AnaDO). AnaDO holds an Analog and Onoff objects hybrid behaviour and it is found in most of the industrial pumps in fields like vacuum, cooling and HVAC.

A detailed list of the basic objects together with their category is presented in the Table 1.

Table 1: UNICOS-CPC basic objects

Name	Function	Category
xI	x=Analog/Digital Input	I/O
xO	x=Analog/Digital Output	I/O
AIR	Analog Input Real	I/O
AOR	Analog Ouput Real	I/O
xPar	x=Analog/Word/Digital parameter	Interface
xStatus	x=Analog/Word/Digital status	Interface
Analog	Valves, Heaters,...	Field
OnOff	Motors, Pumps,...	Field
AnaDig	PWM, slide valves	Field
Local	Local hand valves	Field
AnaDO	Pumps, Frequency Variators	Field
xAlarm	x=Analog/Digital alarm	Control
Controller	PID feedback control	Control
PCO	Process Control Unit	Control

Among all the modifications in the objects, important changes to mention are:

- Full stop functionality in the field objects. Up to now the PCOs could restart once the interlock was acknowledged and their cause had disappeared. In the new version, the control engineer may specify that the device in full stop mode must be explicitly allowed to restart by the operator. This decouples the interlock acknowledge from the operator explicit restart action.
- Controller mode of operation compliant with the UNICOS modes and standardized working states (regulation, output positioning and tracking). The engineer can work with normalized or with physical units. The cascade controller structures can be now generated from specifications.
- Double inverted output in the OnOff objects.
- Addition of a real PWM (Pulse Width Modulation) functionality inside the AnaDig object.
- Alarms on the Local objects expected positions (e.g. a hand valve expected to be closed but its low end-switch is not active)
- Analog alarms thresholds can be parameterized or imposed by the logic code dynamically.

New objects creation by the use of a meta-model

One of the reengineering process tasks was the polishing of existing objects. There were several inconsistencies between the objects in terms of their internal functionalities, interfaces and naming homogeneity. The proper way to homogenize them was the creation of a UNICOS model.

The model is supported by a meta-model describing the properties of the model [5]. The model describes families which are used by the different objects (e.g. *FEDeviceParameters* to define the object parameters). Having such model has both allowed the refinement of the objects and also allowed the framework to create new objects. To support the creation of the objects a specialized tool was developed: the TCT (Type Creation Tool) which permits a drag-and-drop based mechanism to build new objecs and create the device type definition based on XML (eXtended Markup Language).

UAB: Factory automation tools

The previous version of the UNICOS-CPC relied on the automatic generation tools based on Microsft Access. Two different generators were used for each PLC supplier (Schneider and Siemens). The code generated by the tool was somehow attached to the generator itself and the project versioning was difficult.

The importance of appropriate flexible and scalable tools is more visible when dealing with large control systems. Control engineers require them to develop such large systems within a reasonable time. A clear example of this is the LHC cryogenics control where some PLCs are configured using thousands of source code files. Using appropriate factory automation tools optimize the time of application development and increase the engineer performance. Note that one such PLC could deal with more than 2500 I/O channels and 700 field objects (e.g control valves, heaters, etc.)

The need of a more modern and flexible tool dealing with the previous mentioned shortcomings lead to the creation of the UAB (UNICOS Application Builder).

The UAB is formed by different components each one having a specific functionality (See Fig. 1). For example a Siemens S7 plug-in will create the S7 PLC application source code while the SCADA WinCC OA plug-in will be in charge of generating the needed information for the SCADA configuration. A special plug-in, the so-called CPC Wizard, orchestrates the plug-ins involved in creating the control application and shows the control engineer a friendly vision of all the steps to create the process control application [6].

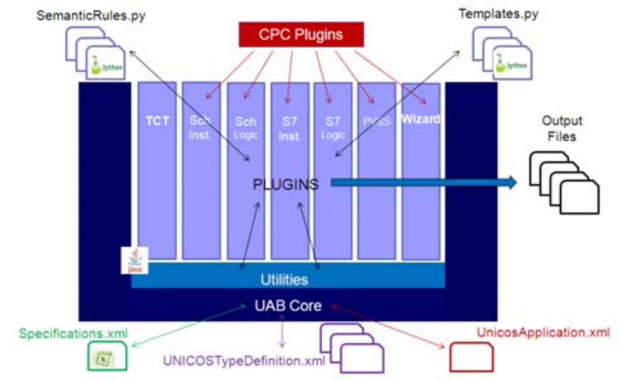

Figure 1 - UAB plug-ins

Another remarkable feature of the UAB architecture is the possibility to include others plug-ins profiting from the already developed core functionalities. This is the case of CoDeSys (Controller Development System), a new plug-in presently being developped which will include a new common IEC-61131-3 based PLC platform.

Control logic: functionality and templates

The control logic templates have been rationalized both, by grouping functionalities which were spread out in several templates and also homogenizing them between the two PLC platforms. Table 2 shows the existing standard control logic templates specifying those which are fully generated by the UAB tool.

Table 2: UNICOS-CPC standard logic templates

Name	Function	User edition
BL	Basic logic	No need
CL	Configuration logic	Yes
IL	Interlock logic	Yes
CDOL	Common Dependent Objet logic	No need
GL	Global logic	Yes
DL	Dependent logic	Yes
SL	Sequencer logic (optional)	Yes
TL	Transition logic (optional)	Yes

The format of the logic templates, where the control engineer fills in the process specific logic, has changed. Previously they were text-like files where PLC programming code was explicitly written together with a special language based on Visual Basic to exploit parameterization. Now, with CPC v6, the templates are Python oriented and naturally exploit the benefits of using a scripting language to generate the final PLC code. The advantage: an effortless way to exploit the parameterization included in the control specifications.

Local operation: large vs. small scale projects

Both large and small industrial facilities have different requirements with respect to their automation and operation. When dealing with large projects, the requirements for operation and/or development are quite restrictive and, usually, operation requires centralized control rooms. On the other hand, often, the operation of medium or small projects is not realized 24/7 in front of dedicated HMIs, rather, a subset of information is communicated permanently to a multi-purpose control room where operators, following a multitude of general services, get this information from various processes (e.g. electricity, cooling, ventilation, etc.). Usually that information is a, reduced but still meaningful, set of alarms which should trigger the attention of the operators who, in turn, ask the experts to deal with a problem in case of need. Following this triggering event, the expert connects to the SCADA to check the details of the problem.

There are also cases where domain experts need to command industrial installations locally, and hence, the need of a local interface: the industrial operation touch panels answer that requirement. Availability is also an issue, because the classical SCADA operation relies on Ethernet networks that provide the supervision servers access to the PLCs. Local panels connect naturally to the PLCs with native industrial fieldbuses (e.g. MPI, Profibus, etc.) and exhibit the needed robustness for use in an industrial environment. The new version of UNICOS CPC includes a new mode of operation called *local*. This mode is activated once the object is taken by means of the local panels deployed in the field. The package incorporates a library of objects (faceplates and widgets) to build applications in local touch panels in the same way of what has been offered in the classical SCADA.

Siemens PLC new architecture

The experience gained with the SIEMENS PLCs in the UNICOS-CPC environment presented a noticeable drawback. In fact, the creation of a new object in the package imposed large modifications to the PLC application architecture. The goals, during the reengineering phase, were to remove this constraint and also to eliminate any required user parameterization once the application is generated.

The usage of the UNICOS model imposed the standardization of the existing and future objects. This, in turn, allows an optimal integration of new objects and an optimal usage of the SIEMENS S7 PLC resources.

Operation using flexible recipes

An important feature of a large control system is the ability to apply the appropriate set of parameters at the time they are pertinent. The previous version of the package already included the capability of defining recipes but with a disadvantage: the need of a priori definition of the recipe components.

A flexible recipe mechanism has been developed in the new package version. Flexible recipe management is recognized as a highly desired capability during operation where the plant operators need to tweak the recipes and adapt their components and values to their requirements without rebuilding the control application. The designed recipes can run in multiple PLCs and still be synchronized in its triggering.

The mechanism is PLC platform independent and it is designed in an open and reusable basics, thus becoming in a SCADA component.

HMI improvements

There have been many modifications concerning visualization in widgets and faceplates: without enumerating the all, there are some new features which deserve attention.

First, because there is a recognized need today to show on the same screens many different control systems (e.g. cooling, HVAC, vacuum, etc.), the colours employed in the HMI had to be redefined. There are no standards in this field, so the engineers must be judicious in their choice. The result is a decrease in the number of employed colours to 4 base colours (green=ok, red=alarm, orange=warning and yellow=force values) and 2 additional error indications (cyan=communication and purple=no data).

Second, the devices can be referred to both with the control name and the expert name. This new feature allows the different experts to have their own project dependant nomenclature in naming their devices.

Third, the alarms representation incorporates the level concept. Following the recommendations of the ANSI/ISA-18.2 standard, a new field on the alarms explicitly characterizes the type of alarm. The four levels employed are fully compatible with the classification of the CERN alarm system.

And four, the objects incorporate two additional navigation options, their positions within the control hierarchy and also the state of the interlocks affecting them in the case of the field and PCO objects.

CONCLUSIONS AND FUTURE PROSPECTS

A large reengineering process has been carried out on the UNICOS CPC package. As a result, CPC v6 provides a better tool suite to automatically create their industrial control applications and also a framework where they can easily extend the objects library in a standard way. The package is already being used in the CERN cooling and HVAC processes.

Following this large reengineering phase, the UNICOS CPC v6 package accepts new capabilities. There are 4 main axes where the tool could develop: auto tuning of controllers, advanced control, redundancy and finally safety systems (See Fig. 2).

Control systems performance depends largely on the correct behaviour of controllers. Most of them are PID (Proportional, Integral and Derivative) and the task of tuning such controllers is widely recognized as a hard job. The package plans to offer automatic tuning mechanisms to the operators.

Figure 2. Future work lines

Some processes show strong nonlinear behaviour and then are intrinsically difficult to regulate. Advanced control solutions are developed to cope with such processes and to help operators. Basic modern control includes a portfolio of solutions from feed-forward, override, cascade to other more sophisticated as Smith predictors and model based predictive control. Some of these solutions are already in the UNICOS CPC v6 (e.g. override, cascade), others will be integrated in the following versions.

Redundancy at the control layer ensures increased availability in the control system. Some processes impose redundancy because of their large downtime in case of PLC failure. UNICOS-CPC will ensure the integration of such components for both Schneider and SIEMENS PLCs.

Finally, safety systems are more often required by the control engineers in processes where safety (personnel and/or equipment) is an issue. Plans for the creation of a similar framework for safety systems are being prepared.

REFERENCES

[1] J. Casas-Cubillos, et al. "Application of Object-Based Industrial Controls for Cryogenics", EPAC'02

[2] Ph. Gayet, et al. "UNICOS a framework to build industry like Control systems: Principles and methodology", ICALEPCS'05, Geneve 2005

[3] D. Willeman, et al. "UNICOS CPC New domains of applications: vacuum and Cooling and Ventilation", ICALEPCS'11, Grenoble, 2011

[4] H. Milcent, et all. UNICOS: An Open Framework. et all. ICALEPCS 09

[5] B. Copy, et al. "Model Oriented Application Generation for Industrial Control Systems", ICALEPCS'11, Grenoble, 2011

[6] B. Fernandez Adiego, et all. "UNICOS CPC6: Automated Code Generation for Process Control Applications", ICALEPCS'11, Grenoble, 2011