UNICOS

Automating the Automation

Enrique Blanco (CERN)
on behalf of the UNICOS team

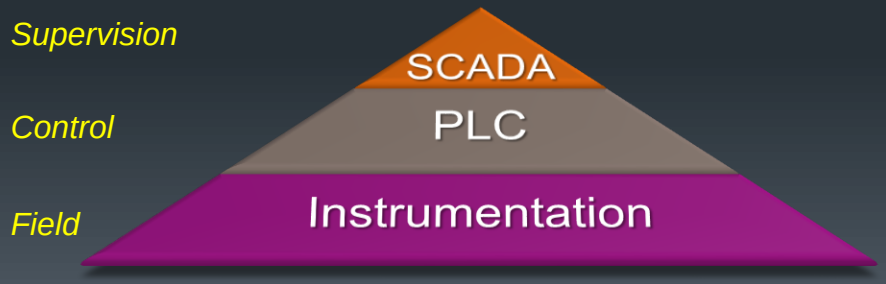
Outline

- UNICOS foundation
- Device model
- Methodology
- Automatic control systems generation
- Conclusions

A look to the past

[1998] **UNICOS** (**UN**ified **I**ndustrial **C**ontrol **S**ystem) was born at CERN as a need to develop the LHC cryogenics control system

The goal was to create an industrial control system covering the upper two layers of the typical **automation** pyramid.

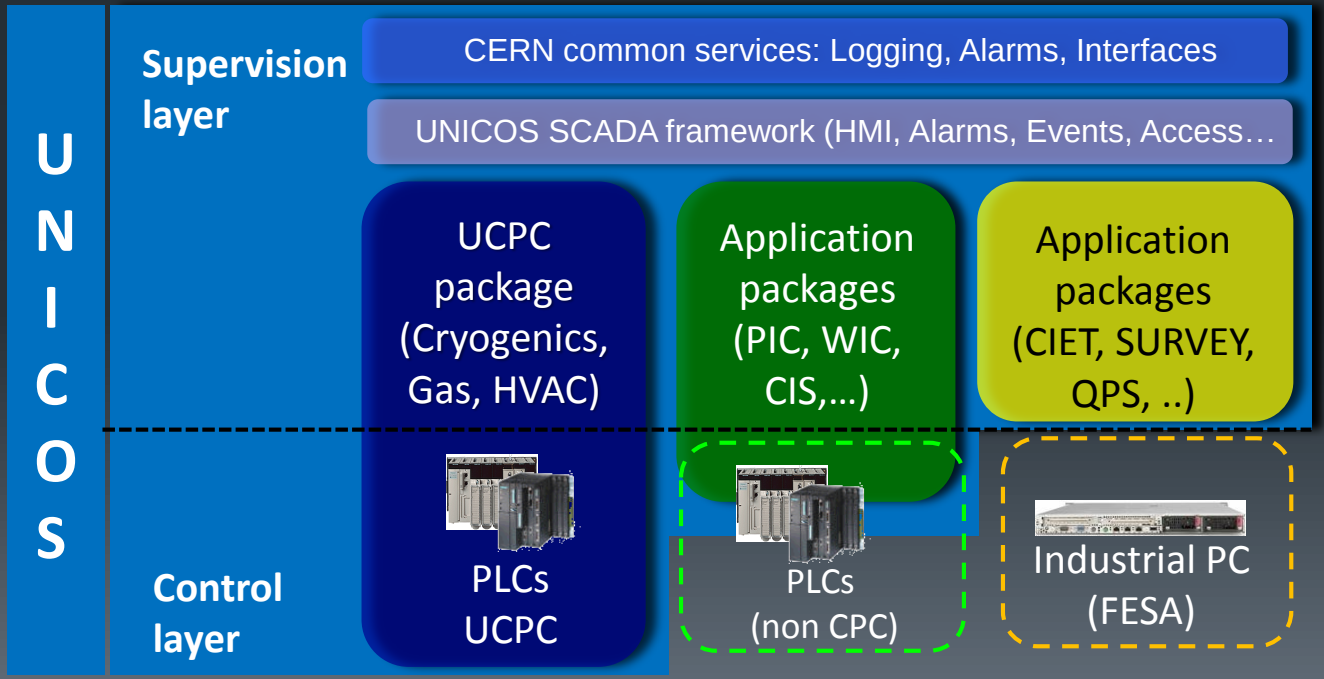


PcVue
SIMATIC WinCC Open Architecture
SIEMENS
Schneider Electric

UNICOS

- UNICOS is a framework to create industrial control applications

  UNICOS CPC: A basic package (**Continuous Process Control**) to develop integrated process control applications.



UNICOS-CPC framework basics

- A collection of standard devices types (objects)
  - CPC: Generic library covering most of the equipment of continuous processes
- Methodology
  - Modeling of the process by control modules based on a decomposition method (ISA-88)
  - A formalized and standard way of programming the specific process logic
- Operation
  - Standard HMI allowing an homogenized operation (navigation, trends, access control…)
  - Suite of standard CERN systems: Alarms, DB Logging, Middleware communications…
  - Diagnose capabilities (process alarms, events, system integrity…)
- Versatile suite of development tools (UAB: UNICOS Application Builder)
  - Automatic instantiation of the devices and logic code

UNICOS CPC applications scope

Process Control

- LHC Cryogenics
- C&V: Cooling and HVAC projects
- LHC Gas Control System
- AWAKE plasma cell

Interlocks

- LHC Collimator Temperature Interlocks
- LHC Test benches facilities
- FAIR magnets testbenchs interlocks

Motion

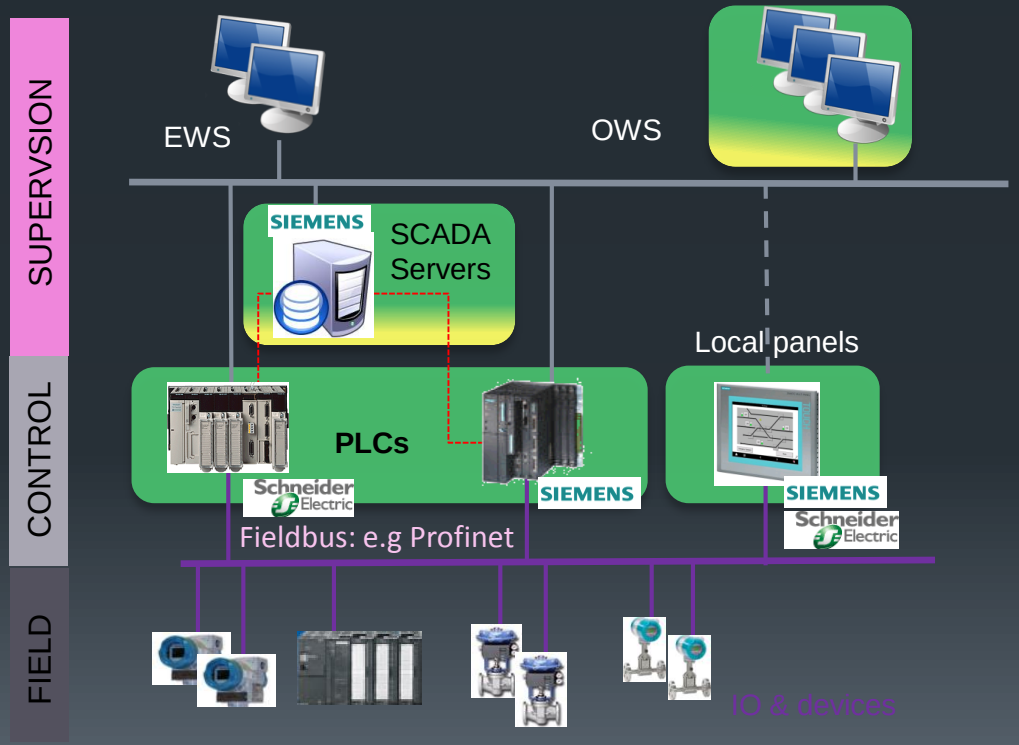
- HTS winding machine
- ATLAS big wheels
- LHC elevators
- AMS beam test servo systems

Vacuum

- LHC Detector vacuum: ATLAS, CMS
- REX vacuum control
- ISOLDE Vacuum control

Industrial Controls Components

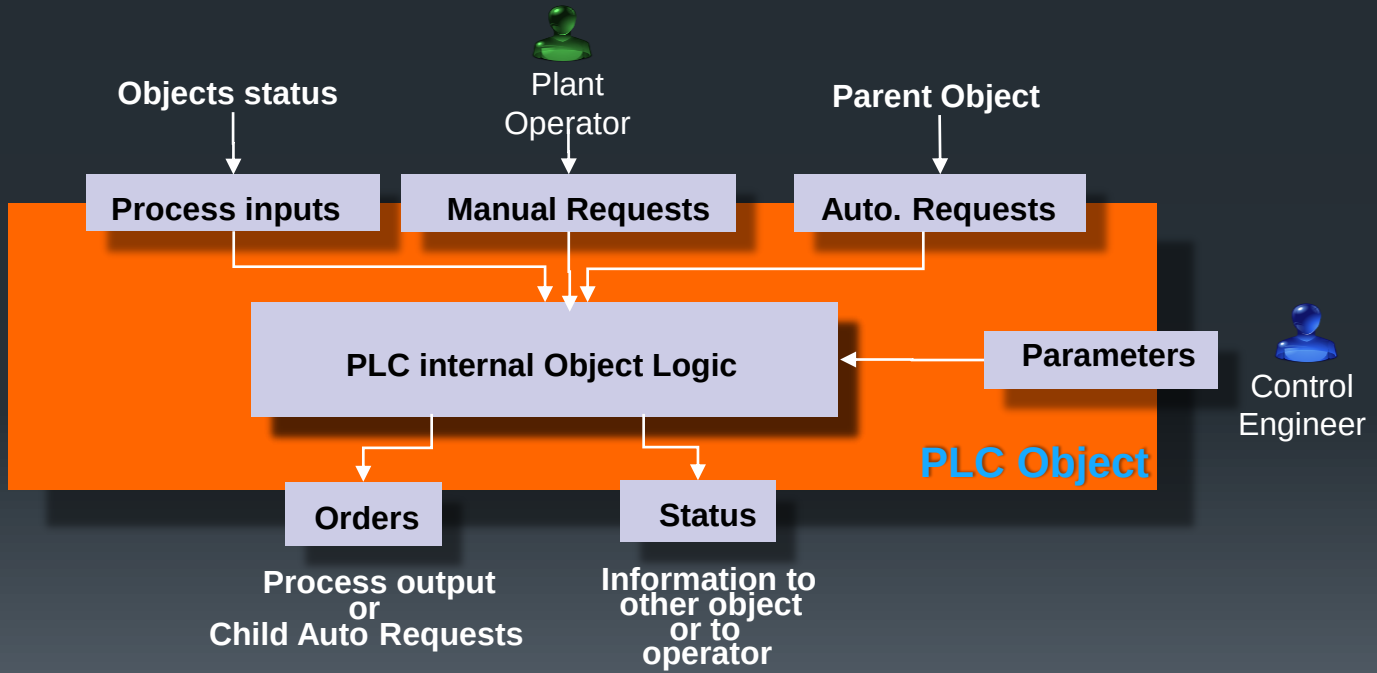
- **UNICOS** relies on industrial off-the-shelf components
  - SCADA: WinCC OA
  - Touch panels: Siemens, Schneider
  - PLCs: Siemens, Schneider, Codesys-based



Outline

- UNICOS foundation
- Device model
- Methodology
- Automatic control systems generation
- Conclusions

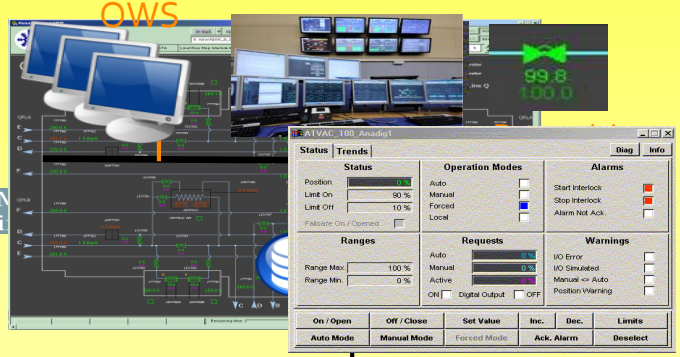
UNICOS CPC Object Model



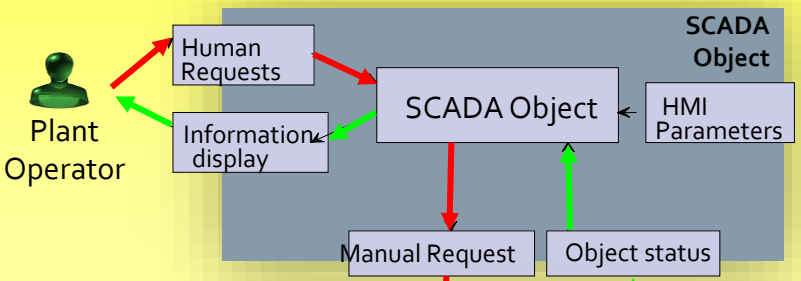
Objects & Layers Integration

Supervision Layer

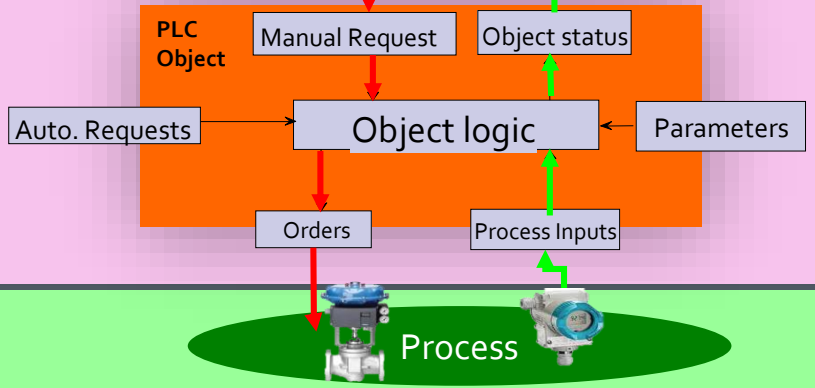
OWS



In the Supervision layer the object presents the relevant information to the operator and allow manual commands



Control Layer



Field Layer

Not only a bunch of objects

- A well defined set of **standard device types** (objects), modeling most of the equipment and needs of continuous processes and the **relationships** between them.
  - **I/O Objects**
    - ✓ Digital I/O
    - ✓ Analog I/O
  - **Field Objects**
    - ✓ OnOff
    - ✓ Analog, AnaDig
    - ✓ Local
    - ✓ AnaDO
  - **Control Objects**
    - ✓ Controller
    - ✓ Alarms
    - ✓ Process Control Object
  - **Interface Objects**
    - ✓ Parameter (Digital, Word, Analog)
    - ✓ Status (Word, Analog)
  - ✓ **Motion**
    - ✓ Stepping Motor
    - ✓ Encoder

- UNICOS CPC provides libraries (control and supervision layers)
- A **formalized** methodology to:
  - **Define** the **control units** of a process (ISA-88 standard: Batch processes)
  - **Programming** the **specific process logic** for those units

A flavor in objects: Field objects

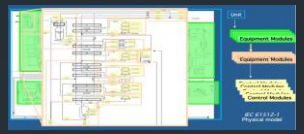
- **Functionality**
  - Model the real field equipments (e.g. pumps, valves…)
- **Types**
  - *OnOff*: Binary Objects
    (e.g. on/off valve, motor, pump)
  - *Analog*: Analog objects
    (e.g. control valve, heater)
  - *Anadig*: Analog inputs and Digital outputs objects
    (e.g. valves/heaters controlled by on/off pulses)
  - *AnaDO : Similar functionality of an OnOff + Analog object*
    (Motor with VFD, Thyristor, Heater, etc.)
  - *Local:* Field localized objects *:*
    (e.g. manual valve)

Outline

- UNICOS foundation
- Device model
- Methodology
- Automatic control systems generation
- Conclusions

UNICOS Engineering life cycle

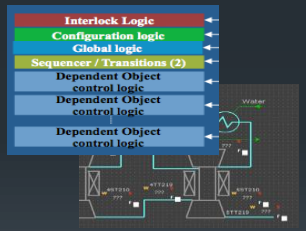
Decomposition



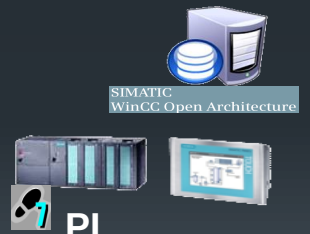
Specifications

Automatic Code Generation

Control Logic specifics & Synoptics

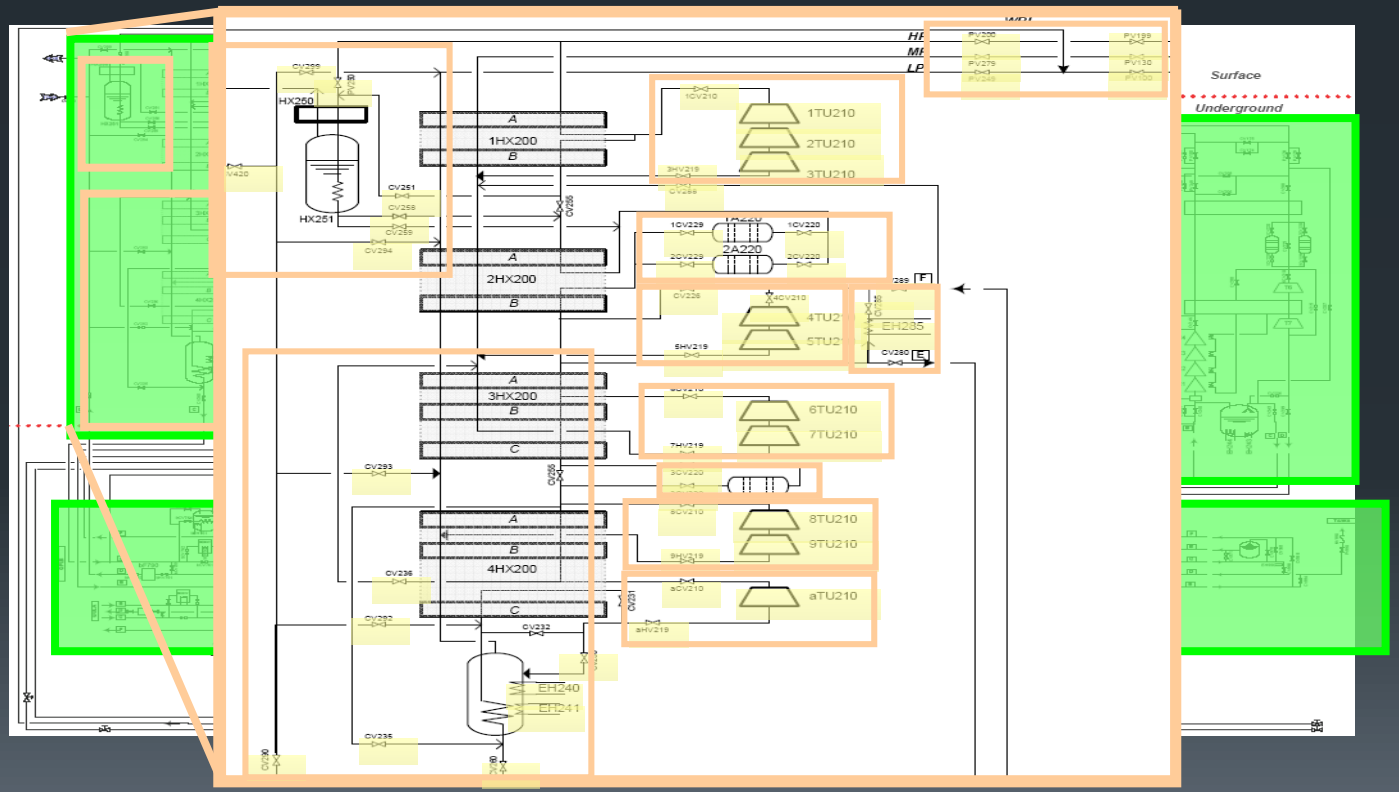


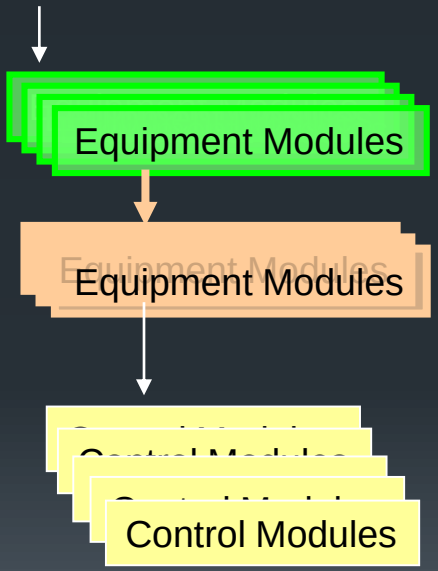
Deployment



Reverse Engineering

Process Analysis: Decomposition





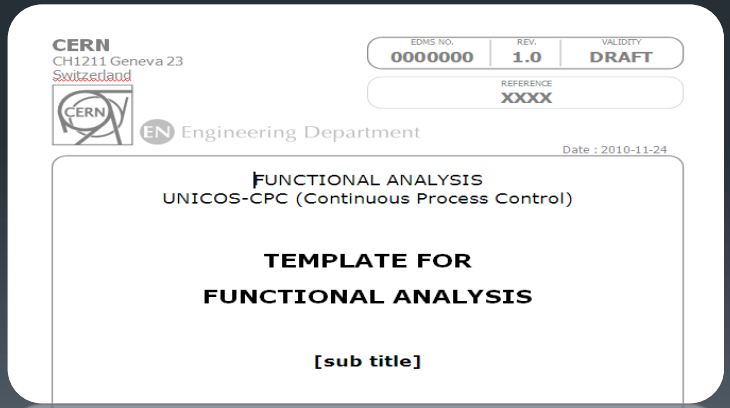
IEC 61512-1
Physical model

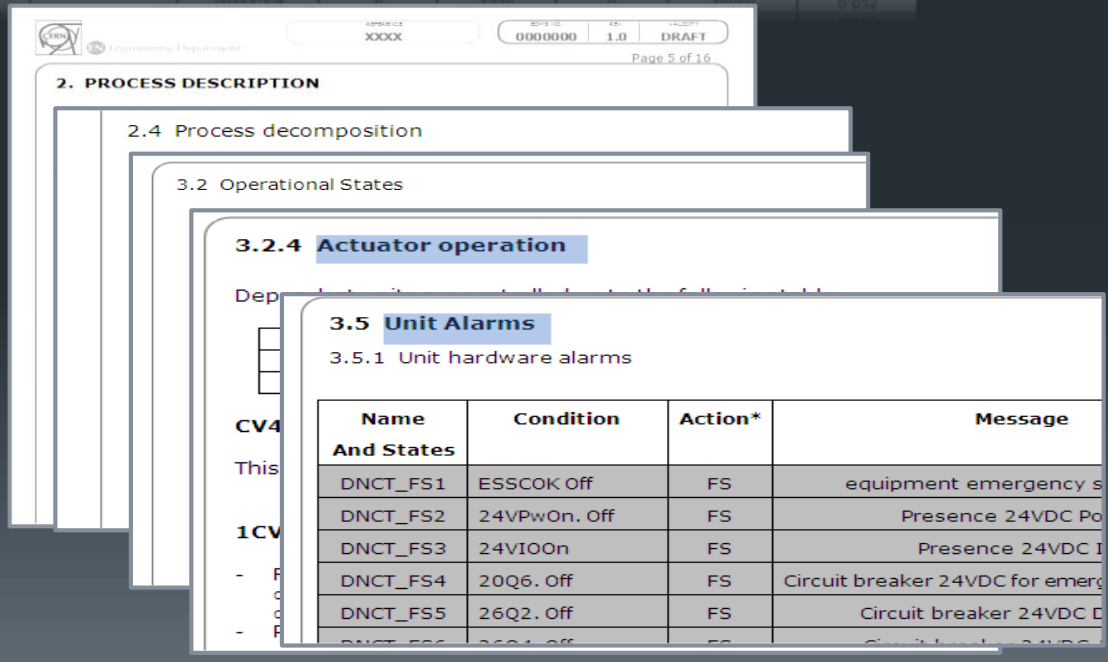
Specifications

UNICOS CPC Specs (xls/xml file)

DeviceIdentification	DeviceDocumentation			FEDeviceIOConfig		FEDeviceParameters				
					FEChannel					
Name	Description	Electrical Diagram	Remarks	FE encoding type	InterfaceParam1	Range Min	Range Max	Raw Min	Raw Max	DeadBand (%)
QSDN_4_1TT4001	Vessel 1- Heater section1-Temp. control	AI1.0			%IW1.1.0	80	350	0	10000	0.025
QSDN_4_AI1	SPARE	AI1.1			%IW1.1.1	0	100	0	10000	0.025
QSDN_4_1TT4002	Vessel 1- Heater section2-Temp. control	AI1.2			%IW1.1.2	80	350	0	10000	0.025
QSDN_4_1TT4003	Vessel 1- Heater section3-Temp. control	AI1.3			%IW1.1.3	80	350	0	10000	0.025
QSDN_4_1LE400	Vessel 1- LN2 Level	AI1.4			%IW1.1.4	0	1350	0	10000	0.025
QSDN_4_1PT400	Vessel 1- LN2 Vessel Pressure	AI1.5			%IW1.1.5	0	4.0	0	10000	0.025

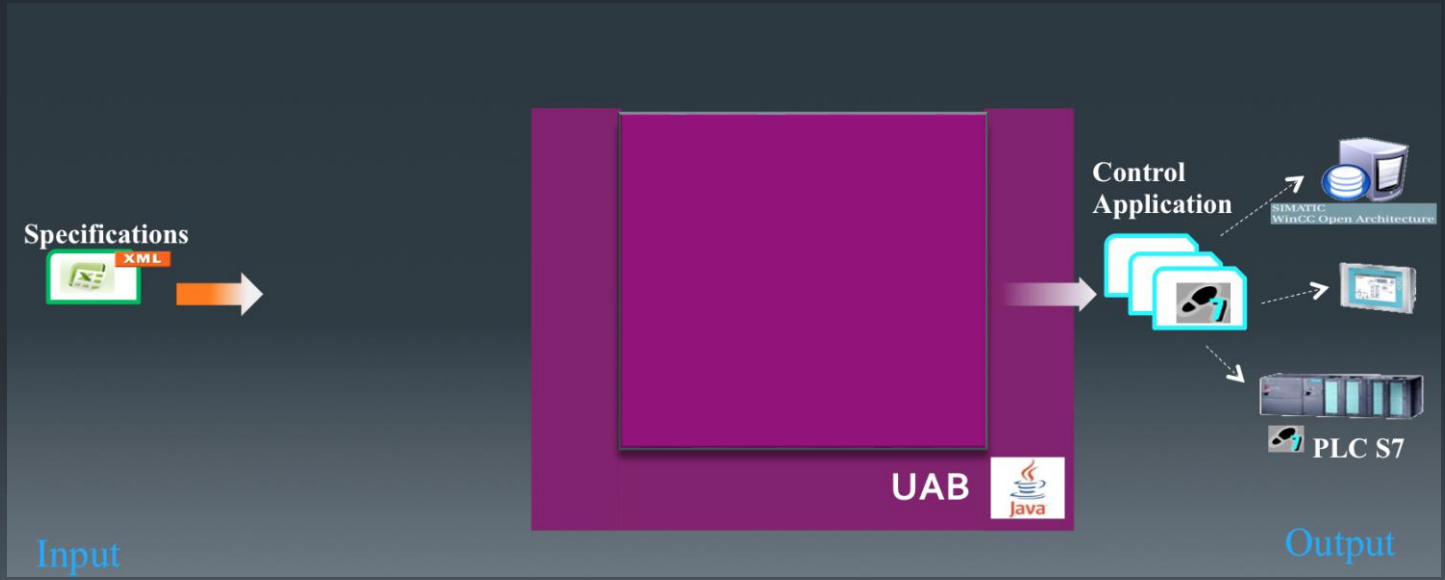
Functional Analysis + Logic specification (Word templates)





Automatic code generation

- In a minute…

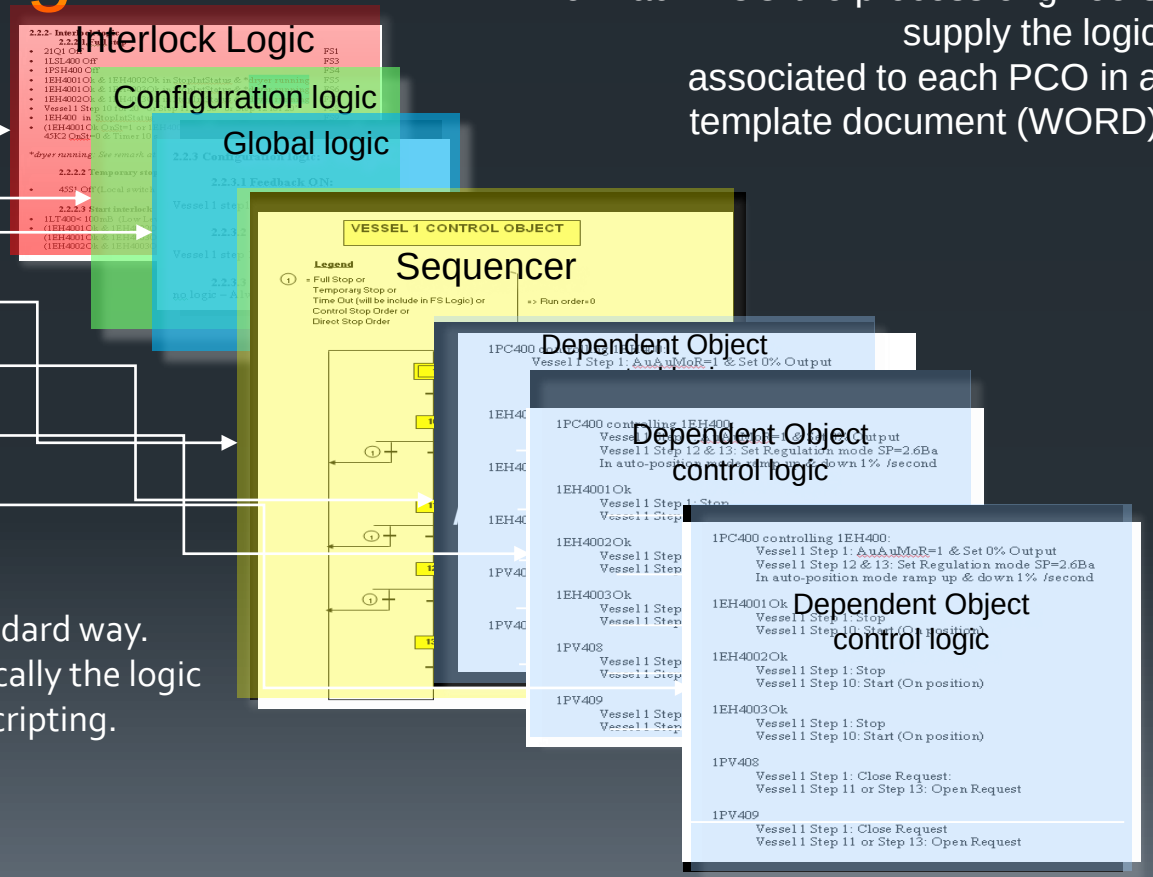


Process control logic

For Each PCO the process engineers supply the logic associated to each PCO in a template document (WORD)

Logic Placeholders

- Interlock Logic
- Configuration logic
- Global logic
- Sequencer / Transitions (2)
- Dependent Object control logic
- Dependent Object control logic
- Dependent Object control logic

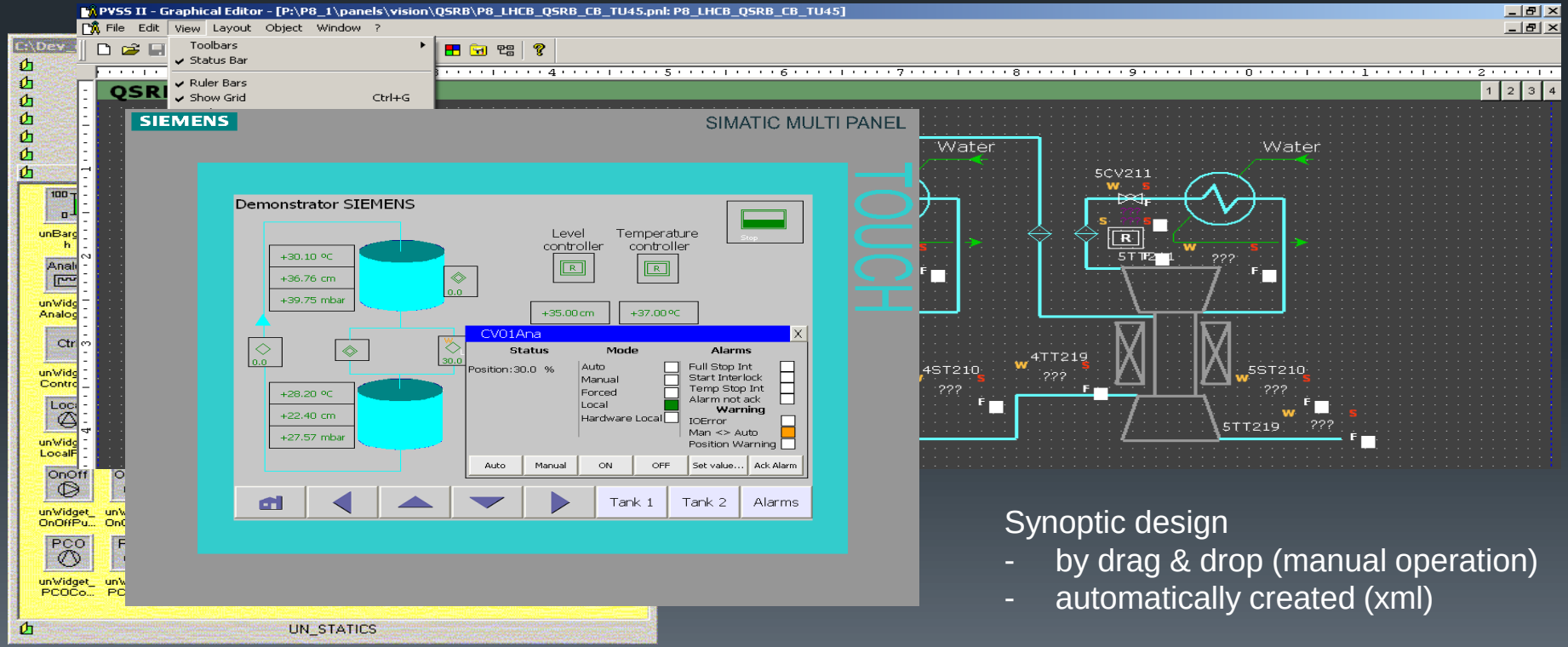


Process logic can be either:

- coded by the control engineer in an standard way.
- some applications may create automatically the logic based on templates based on **Python** scripting.

HMI synoptics

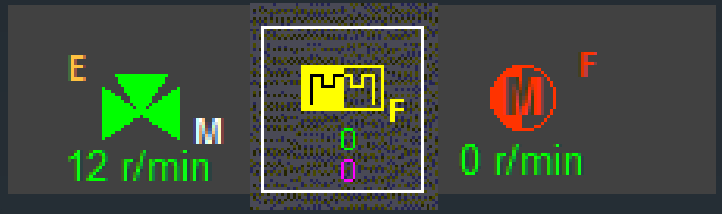
- Manual intervention (or automatic if known a priori)



Synoptic design
- by drag & drop (manual operation)
- automatically created (xml)

UNICOS HMI

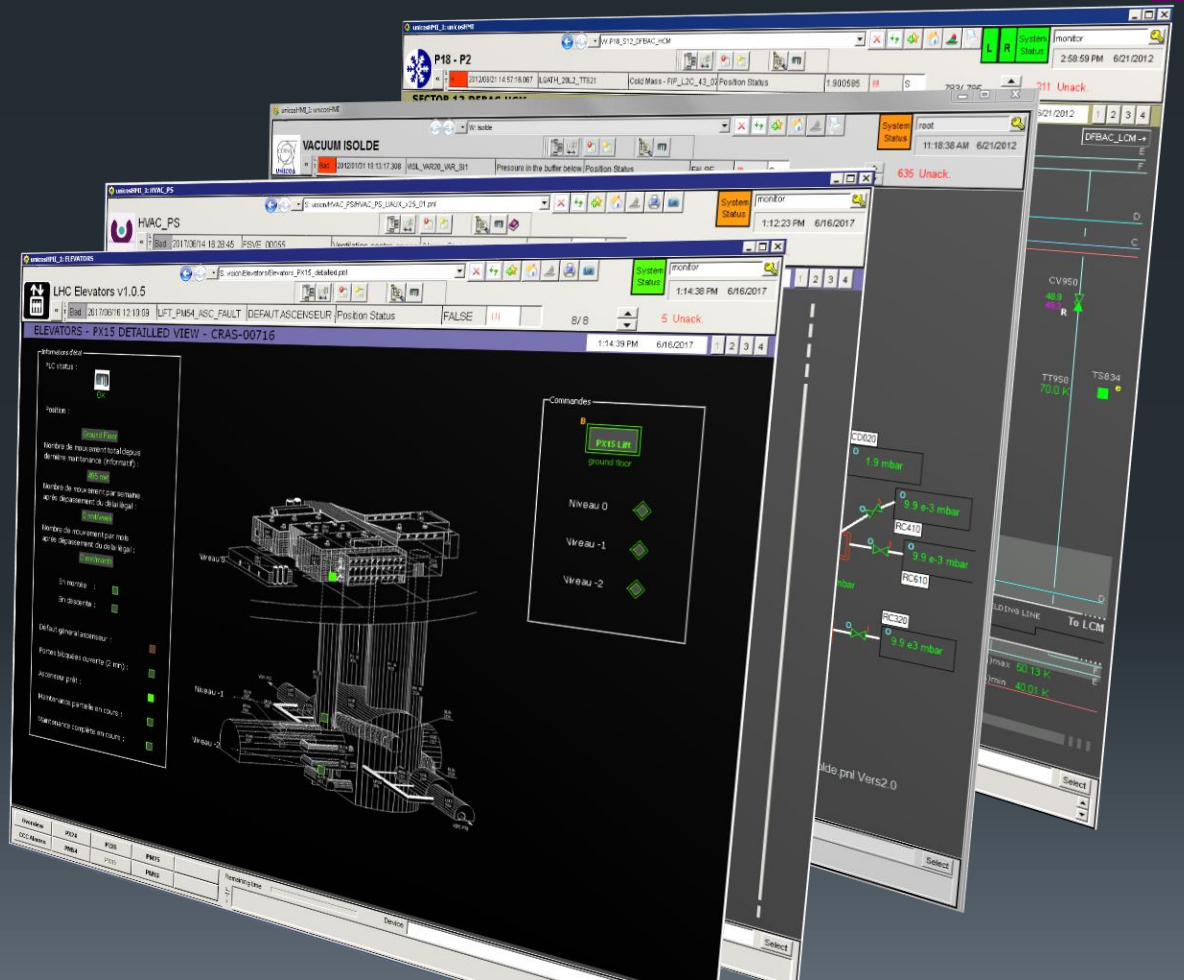
Faceplate



Widgets

UNICOS HMI

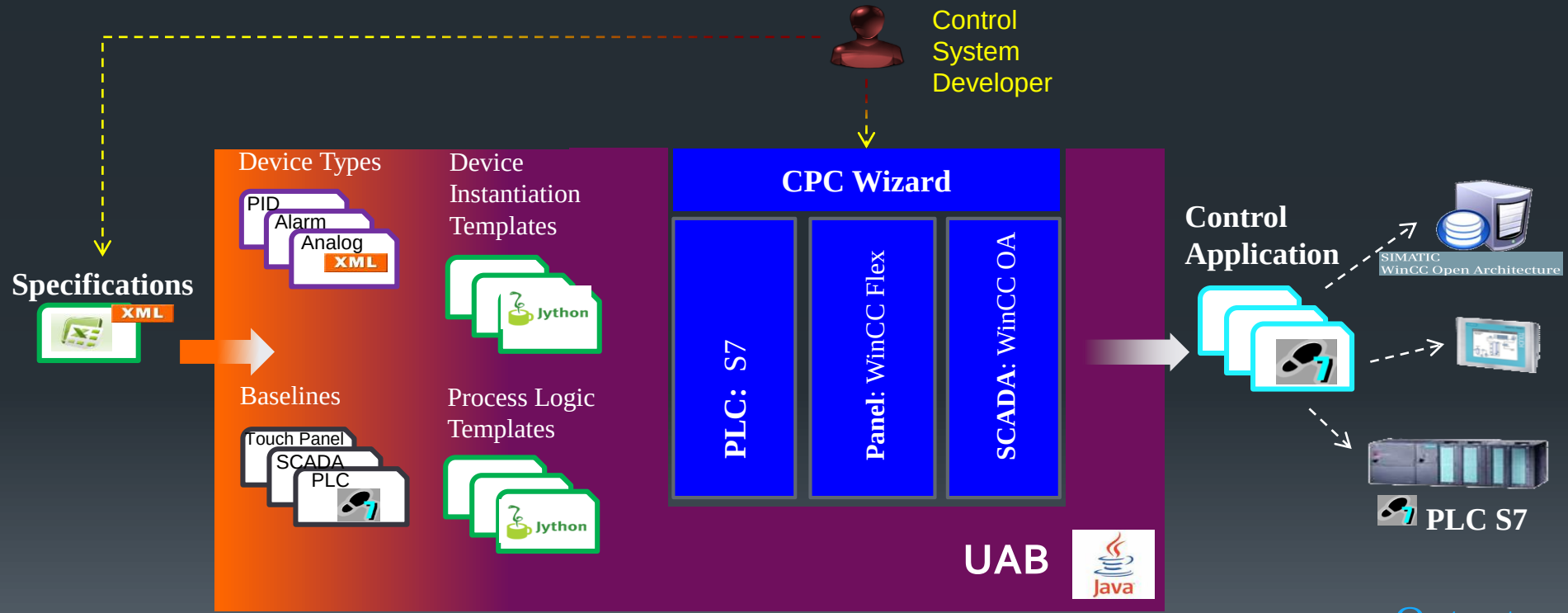
- HMI
- Alarms
- Events
- Navigation
- Trends
- System integrity
- BEEP
- Access control
- Contextual buttons
- Uniform look & feel



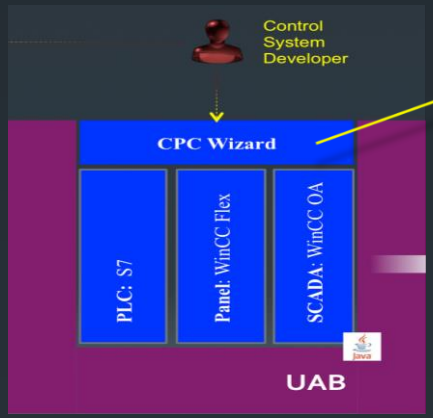
Outline

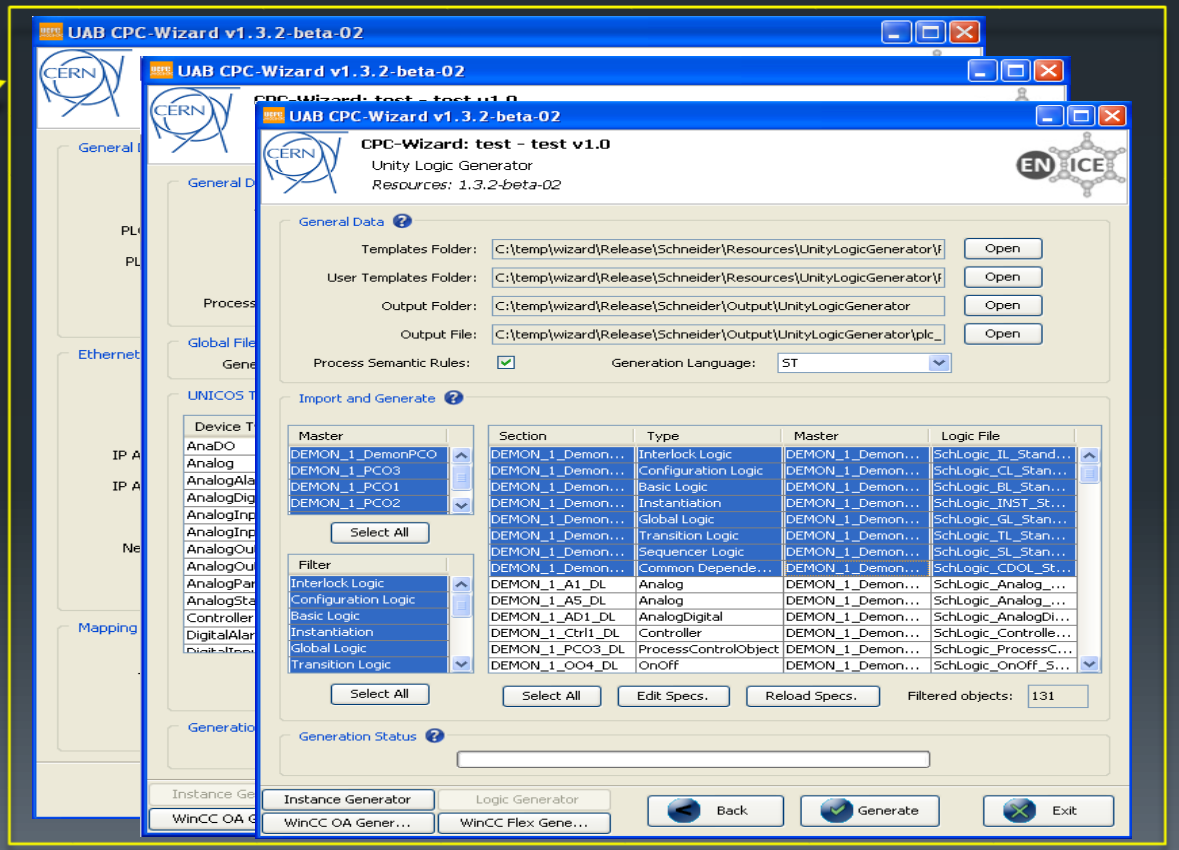
- UNICOS foundation
- Device model
- Methodology
- Automatic control systems generation
- Conclusions

UAB: UNICOS Application Builder



Workflow based on wizards





Outline

- UNICOS foundation
- Device model
- Methodology
- Automatic control systems generation
- Conclusions

UNICOS benefits

- Proven industrial technologies
  - Standard CERN PLC suppliers: Siemens, Schneider (PLCs and Touch Panels)
  - Standard CERN SCADA (WinCC OA)
- Standardization
  - Based on industrial standards: ISA-88 / IEC-61512: Batch control
  - Uniform and maintainable code (IEC languages is not enough)
  - Optimized maintenance and development backup with a central support
  - Same look & feel (and functionality) optimizes operation in the control room
- Rapid development
  - Automatic generation of applications
  - Early commissioning availability. No SCADA development

Conclusions

http://www.cern.ch/unicos

UNICOS framework composed of

- Generic set of reusable devices
- Analysis and development method
- Programming structure
- A rich functionally in a homogenized HMI

Facilitate the task of the automation engineer by allowing him/her in focusing only in the automation duty and not in the software production itself: Automatic generation of code.