UNICOS CPC6: AUTOMATED CODE GENERATION FOR PROCESS CONTROL APPLICATIONS

B. Fernández Adiego, E. Blanco Viñuela, I. Prieto Barreiro (CERN, Geneva, Switzerland)
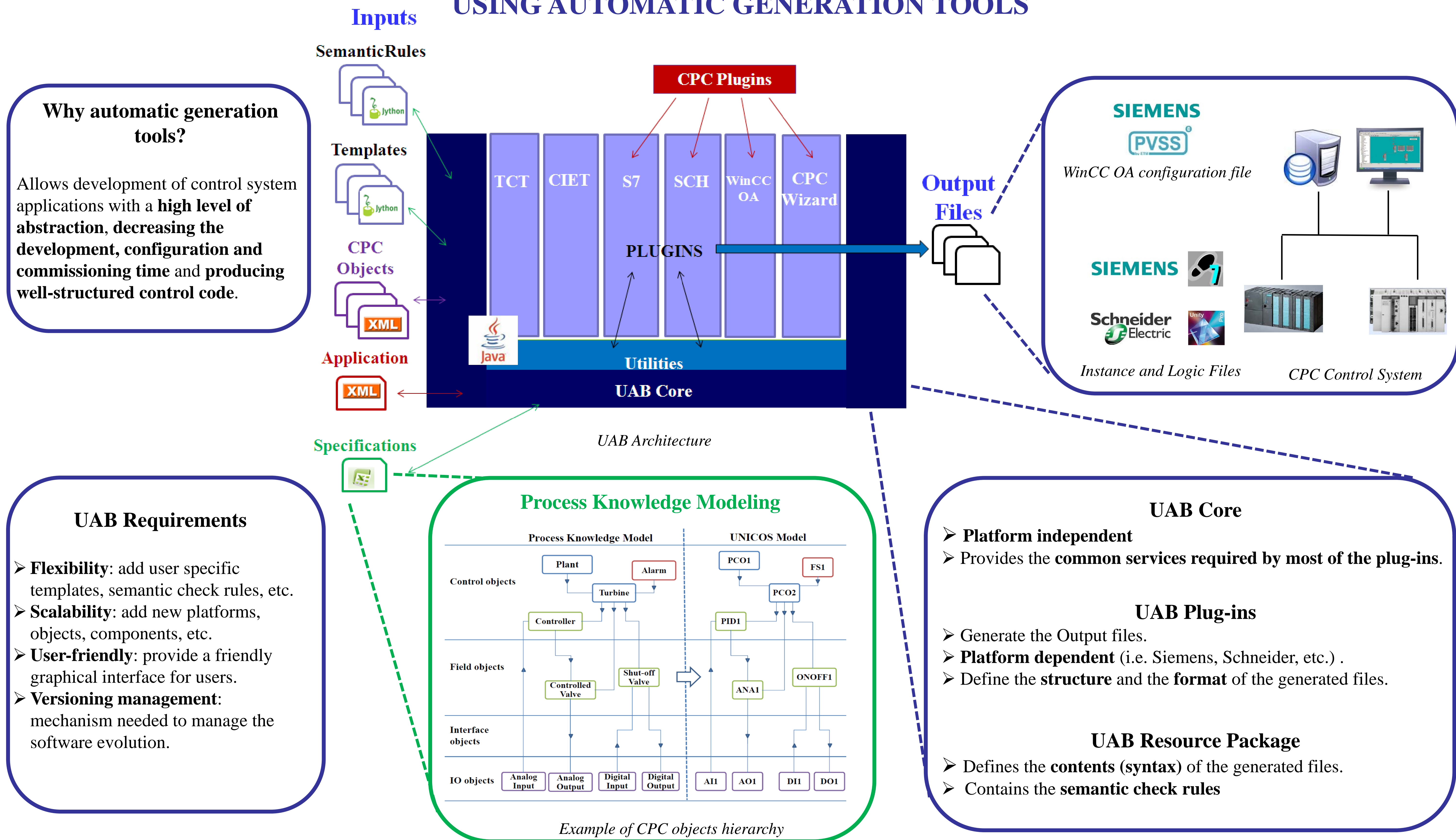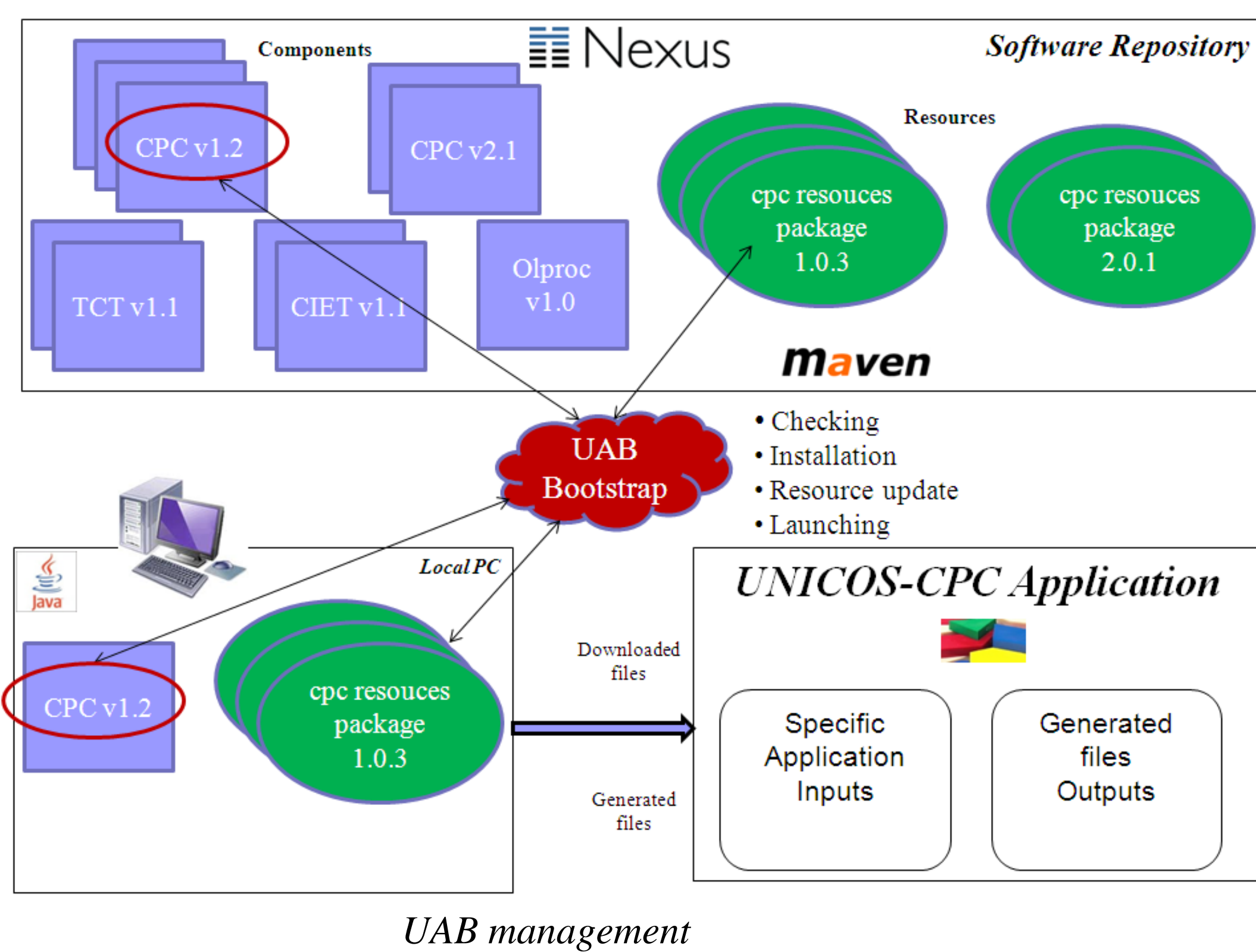
CERN

icalepcs 2011

ABSTRACT

This poster presents the new **UAB–CPC6** (UNICOS Application Builder – Continuous Process Control) component as a part of the UNICOS framework. UNICOS-CPC provides **a well defined library of objects, a methodology and a set of tools** to design and implement industrial control applications. For this new version, one of the most important requirements was to include a more flexible, scalable, consistent and user-friendly automatic generation tool. As result of these requirements the **UAB** has been developed. This tool generates the control code extracting information from the UNICOS objects definition and the process knowledge included in the specification file. Important benefits for CPC users and developers have been achieved.

USING AUTOMATIC GENERATION TOOLS

Inputs

SemanticRules

Templates

CPC Objects

Application

Specifications

UAB Architecture

CPC Plugins

| TCT | CIET | S7 | SCH | WinCC OA | CPC Wizard |

PLUGINS

Utilities

UAB Core

Output Files

SIEMENS PVSS
WinCC OA configuration file

SIEMENS
Schneider Electric
Instance and Logic Files

CPC Control System

Why automatic generation tools?

Allows development of control system applications with a **high level of abstraction**, **decreasing the development, configuration and commissioning time** and **producing well-structured control code**.

UAB Requirements

➢ **Flexibility**: add user specific templates, semantic check rules, etc.
➢ **Scalability**: add new platforms, objects, components, etc.
➢ **User-friendly**: provide a friendly graphical interface for users.
➢ **Versioning management**: mechanism needed to manage the software evolution.

Process Knowledge Modeling

Example of CPC objects hierarchy

UAB Core

➢ **Platform independent**
➢ Provides the **common services required by most of the plug-ins**.

UAB Plug-ins

➢ Generate the Output files.
➢ **Platform dependent** (i.e. Siemens, Schneider, etc.) .
➢ Define the **structure** and the **format** of the generated files.

UAB Resource Package

➢ Defines the **contents (syntax)** of the generated files.
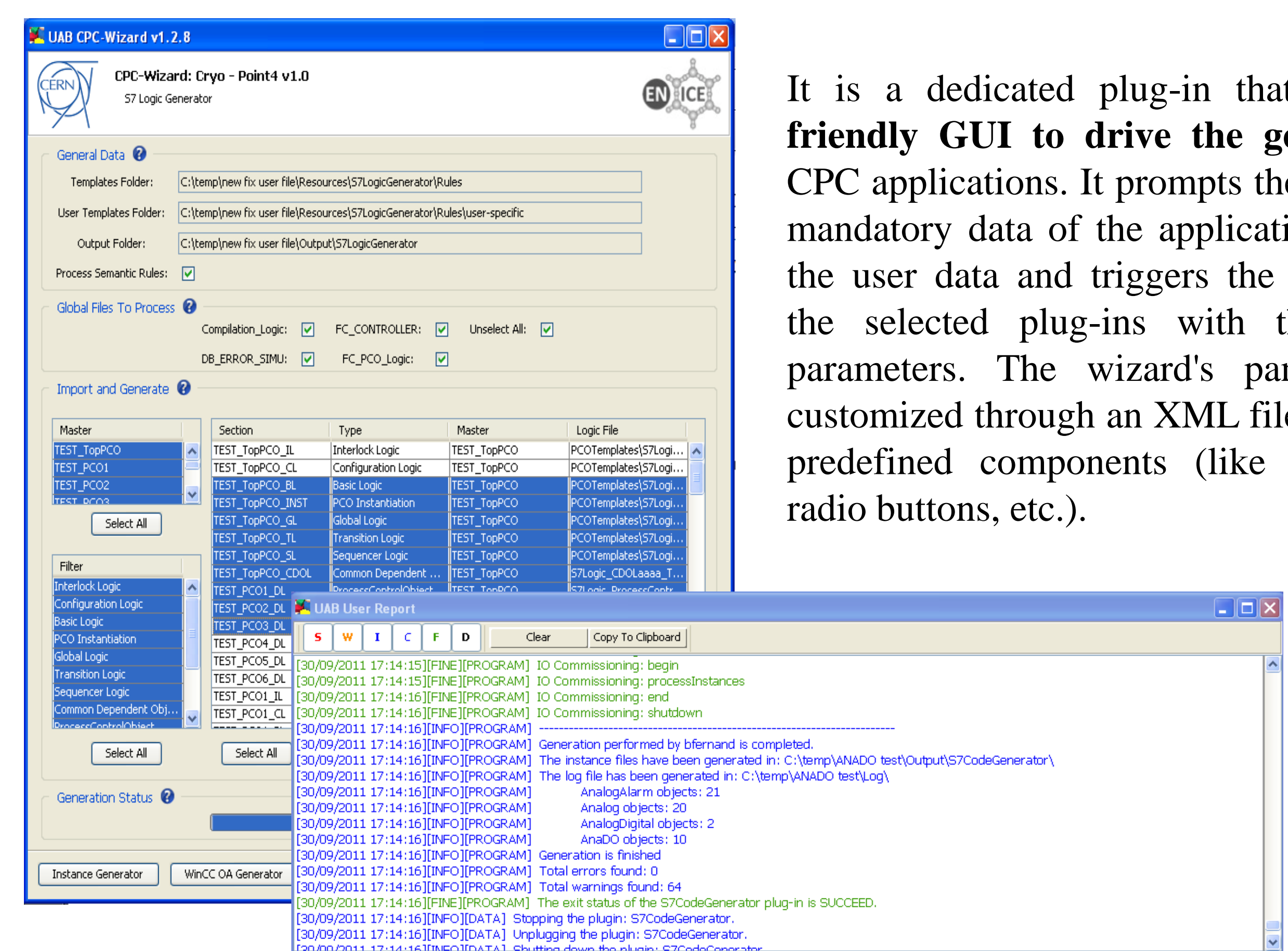➢ Contains the **semantic check rules**

UAB management

The special architecture of UAB, where the software is split in several packages with **different life cycles,** requires an appropriate mechanism to manage the different software versions. The **UAB Bootstrap** is used to manage the different UAB components and its resources packages.

UAB management

CPC Wizard

It is a dedicated plug-in that provides a **friendly GUI to drive the generation** of CPC applications. It prompts the user for the mandatory data of the application, validates the user data and triggers the execution of the selected plug-ins with the specified parameters. The wizard's panels can be customized through an XML file and a set of predefined components (like text editors, radio buttons, etc.).

CPC Wizard and User Report

CONCLUSIONS

The UAB CPC6 component has been developed to add several improvements for developers and users (control system engineers):
➢ The **developer**, using UAB, will obtain more flexibility and performance, **allowing to add new CPC objects easily, new plug-ins for new platforms, templates or semantic check rules.**
➢ For the **user**, this new version offers a **user-friendly graphic interface** to develop CPC applications and a set of services like a **powerful version management** which guaranties the maintenance of these applications, logging, semantic check verifications, etc. The user can also add new objects, templates or check rules to the resources package.