

Towards DB Driven Specifications for Industrial Control Systems

Motivation, work done and share of experience

Diogo Monteiro, Controls Engineer at CERN

PBCS Workshop – ICALEPCS'23

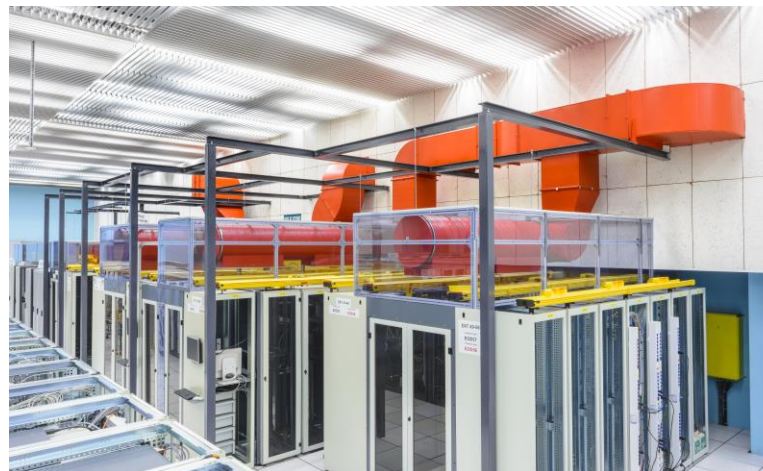
7/10/2023

Cooling and Ventilation Group at CERN (EN-CV)

Cooling

Fluid Distribution

Air Conditioning



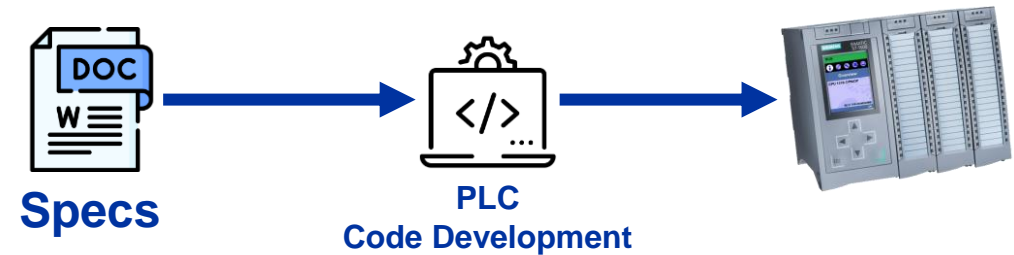
Controls infrastructure for CERN's cooling & HVAC

- **Number of PLC applications: > 650**
 - Split between Schneider and Siemens PLCs
 - (+ some residual legacy hardware)
- **New PLC applications/year: +30**
- **Team's mandate:**
 - Controls development (specification, testing and commissioning)
 - Supervise PLC code development (mostly outsourced)
 - Controls maintenance along plant's life-cycle
 - Management of controls' assets



Specifications for Industrial Control Systems 1/2

- **Main input for the development & operation of control systems**
- **It (usually) contains:**
 - Key information about the controlled process
 - Actuators
 - Sensors
 - Regulation loops
 - Definition of functional decomposition
 - Description of the system's automatic behavior
 - Specification of alarms & interlocks
 - etc



Specifications for Industrial Control Systems 2/2

- **Specifications require about 1/3 of the controls development effort (!)**
- **Some pains we experience:**
 - Readers have heterogenous background: needs common language
 - Low-value elements take a good share of the effort (e.g. formatting)
 - Standardization is difficult
 - Document is hardly exploitable for automatic code generation
 - Version comparison is not user-friendly

Is there an alternative? Let's look at DB-driven specs!

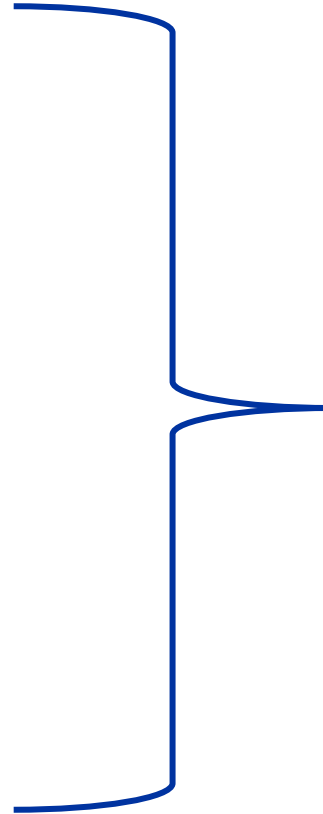
Database Driven Specifications

Main expectations:

- ✓ Structured
- ✓ Content focused
- ✓ Ideal for code generation tooling

... but, we still need:

1. User-friendly editor
2. Text based output

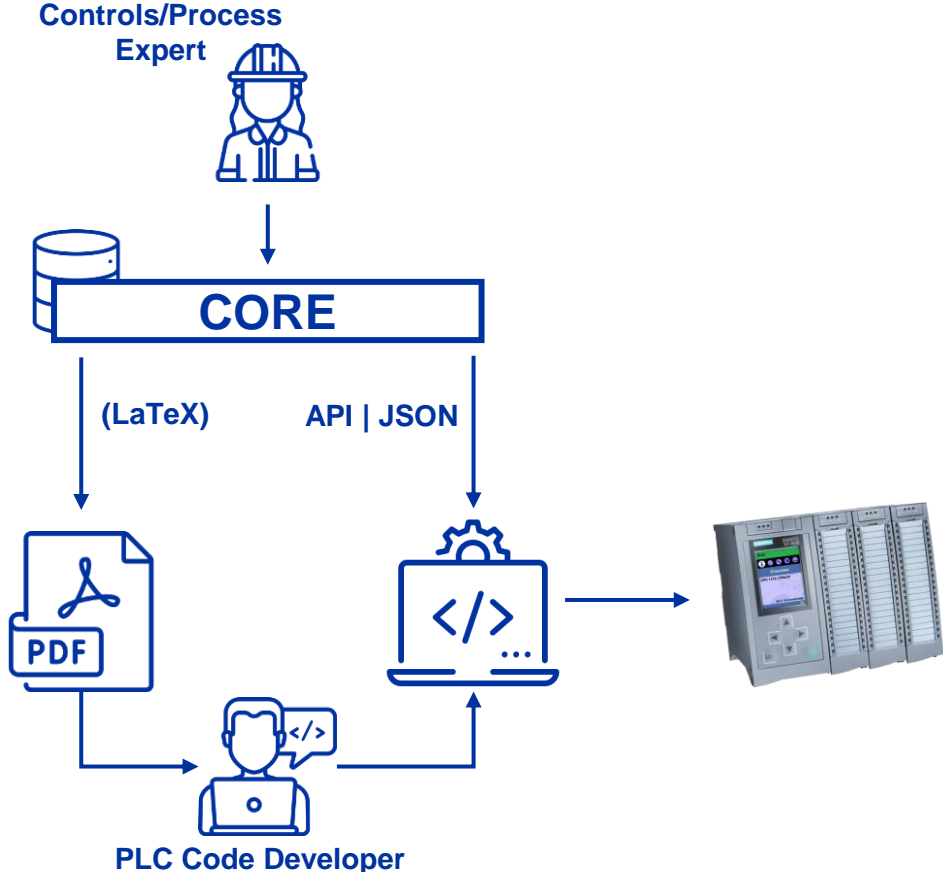


CORE

an in-house tool for DB driven specifications

CORE – a DB driven specifications tool at CERN 1/2

Overview of workflow:



Project timeline & resources:



CORE – a DB driven specifications tool at CERN 2/2

First positive impressions:

1. Less time in formatting, **more time in content**
2. **Standardization** is straightforward
3. Enforced structured -> **focus on the essential**
4. Facilitates **automatic code generation**

... and **challenges to tackle:**

1. Reduced flexibility
2. Parts of specification remain difficult to structure
3. Steeper learning curve for newcomers
4. Resources required for tool development & maintenance

CORE – Sneak Peek 1/3

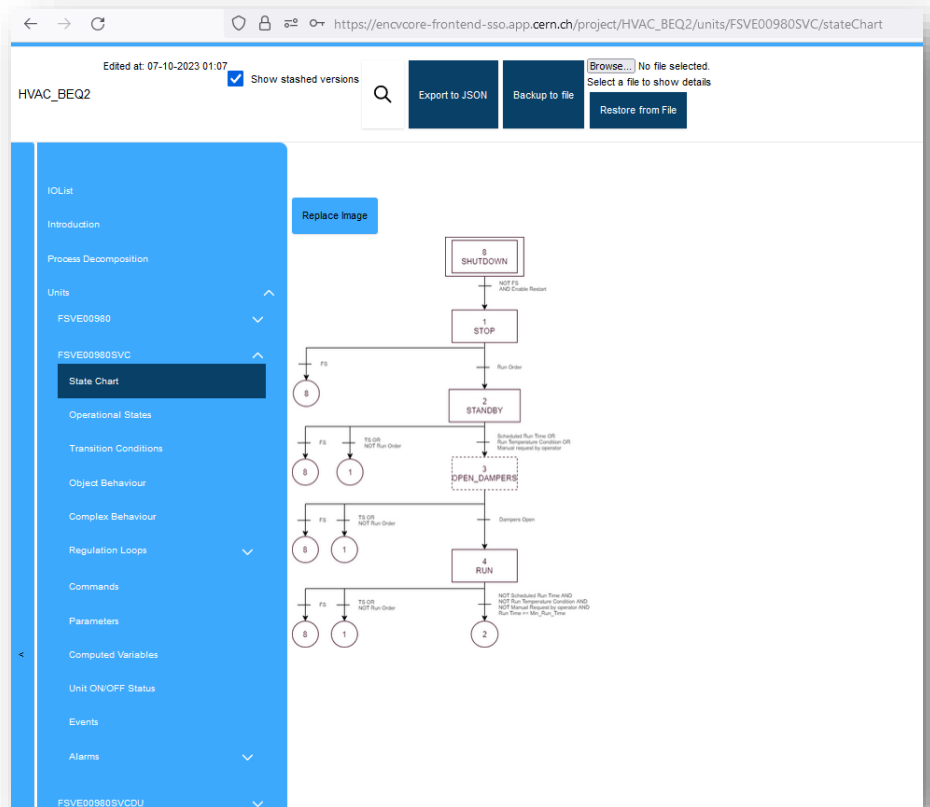
Functional decomposition for plant's control system

The screenshot shows the CORE web application interface for the project 'HVAC_SU1_GASEXT'. The browser address bar indicates the URL: https://encvcore-frontend-ss0.app.cern.ch/project/HVAC_SU1_GASEXT/process-decomposition. The running version is 0.116.0, released on 2023-10-03 at 10:07. The interface includes a 'Reload' button, a search bar, and a 'Show stashed versions' checkbox. The main content area displays a hierarchical process decomposition graph for 'FSVE02180EXT'. The graph shows a root node 'FSVE02180EXT' which branches into 'FSVE02180EXTD' and 'UAUV00069_UUVX20000 ()'. 'FSVE02180EXTD' further branches into 'UAUV00067_UUVS11000 ()' and 'UAUV00068_UUVS12000 ()'. A left sidebar contains navigation links: 'IOList', 'Introduction', 'Process Decomposition', 'Units', 'Supervision', and 'References'. The 'Process Decomposition' section is currently active.

The screenshot shows the CORE web application interface for the project 'HVAC_BEQ2'. The browser address bar indicates the URL: https://encvcore-frontend-ss0.app.cern.ch/project/HVAC_BEQ2/process-decomposition. The running version is 0.116.0, released on 2023-10-03 at 10:07. The interface includes a 'Reload' button, a search bar, and a 'Show stashed versions' checkbox. The main content area displays a hierarchical process decomposition graph for 'FSVE00980 BEQ2'. The graph shows a root node 'FSVE00980 BEQ2' which branches into 'FSVE00980SVC SVC_CV Ventilation R-402 and R-403' and 'FSVE00980SW SWITCH Ventilation R-404'. 'FSVE00980SVC SVC_CV' further branches into 'FSVE00980SVCDCU DIST Double AHU' and 'FSVE00980_AQE13001 ()'. 'FSVE00980SVCDCU' branches into 'UUV200366 UUV2-366' and 'UUV200367 UUV2-367'. 'FSVE00980_AQE13001 ()' branches into 'FSVE00980_AQE10001 ()' and 'FSVE00980_AQE13000 ()'. 'FSVE00980SW SWITCH' branches into 'UUV200365' and 'FSVE00980_AQE20001 ()'. 'UUV200365' branches into 'UUV200365_AQE20000 ()' and 'UUV200365_UUVS20000 ()'. A left sidebar contains navigation links: 'IOList', 'Introduction', 'Process Decomposition', 'Units', 'Supervision', and 'References'. The 'Process Decomposition' section is currently active.

CORE – Sneak Peek 2/3

Specification of sequential function charts (SFC)



The screenshot shows the 'Operational State Definitions FSVE00980SVC' interface. It features a table with the following data:

STEP	NAME	DESCRIPTION
8	SHUTDOWN	Fail-safe state of function. State gets activated following Full Stop (FS) interlock.
1	STOP	Function is at rest. State gets activated following Temporary Stop (TS) interlock or stop request.
2	STANDBY	Function is ready to run as long as one of the required conditions is verified.
3	OPEN_DAMPERS	Transitory state to open common function dampers before jumping to run and trigger AHU start.
4	RUN	Nominal operating state of the function. AHU are required to run.

CORE – Sneak Peek 3/3

List of alarms & interlocks

The screenshot shows the 'Unit Alarms FSVE00980SVC' page in the CORE application. The browser address bar indicates the URL: https://encvcore-frontend-ss0.app.cern.ch/project/HVAC_BEQ2/units/FSVE00980SVC/alarms/unit. The page is edited at 07-10-2023 01:07. A sidebar on the left contains navigation links: IOList, Introduction, Process Decomposition, Units (selected), FSVE00980, FSVE00980SVC, State Chart, Operational States, Transition Conditions, Object Behaviour, Complex Behaviour, Regulation Loops, Commands, Parameters, Computed Variables, Unit ON/OFF Status, Events, Alarms, and Unit (highlighted). The main content area displays a table of unit alarms:

UNIT ALARM NAME	CONDITION	CODE	TEMPORISATION	MESSAGE	ACTION	CCC ALARM	AUTO ACKNOWLEDGE
FSVE00980_TT10000_AL	Valid reading of TT10000 (NOT IOERROR) AND value thresholds: - HH: 35 C - H: 32 C - L: 15 C - LL: 12 C		300	PROBLEME TEMPERATURE AMBIANTE SALLE STATION CV (R-402)	AL	FSVE00980_CCC_MAJ	false
FSVE00980_TT13000_AL	Valid reading of TT13000 (NOT IOERROR) AND value thresholds: - HH: 35 C - H: 32 C - L: 15 C - LL: 12 C		300	PROBLEME TEMPERATURE AMBIANTE SALLE SVC (R-403)	AL	FSVE00980_CCC_MAJ	false
FSVE00980SVC_AL1	Time in OPEN_DAMPERS	OPEN_DAMPERS.X	120	PROBLEME TEMPS D'OUVERTURE REGISTRES D'EXTRACTION ET SOUFFLAGE	FS	FSVE00980_CCC_MAJ	false
FSVE00980SVC_AL2	Unit FSVE00980SVCDDU not available	FSVE00980SVCDDU_Dispo = False	0	PCO GESTION VENTILATION INDISPONIBLE	TS	FSVE00980_CCC_MAJ	false

Conclusion

- **MS Word based specs have blocking drawbacks:**
 - Diverges author attention from the essential
 - Any imposed structure is easily breakable
 - Standardization is difficult
 - It's not adapted to automatic code generation
- **Our team is exploring DB-driven specifications as an alternative:**
 - Content oriented
 - Unbreakable structure
 - Facilitates automatic code generation via API/JSON
- **Some challenges remain:**
 - In-house solution required: resources & infrastructure
 - Not all elements are easily “structurable”

Questions ?



home.cern