

# Status of the NSRC SOLARIS Control System

W.T. Kitka, M. Burzyński, M.K. Fałowski, P. Gałuszka, K. Kędroń, A. Kisiel, G.W. Kowalski, P. Kurdziel, M. Ostoja-Gajewski, P. Sagało, M.J. Stankiewicz, T. Szymocha, A.I. Wawrzyniak, K. Wawrzyniak, I.S. Zadworny



UNIWERSYTET JAGIELLOŃSKI W KRAKOWIE

National Synchrotron Radiation Centre SOLARIS, Jagiellonian University, Czerwone Maki 98, 30-387 Kraków, Poland

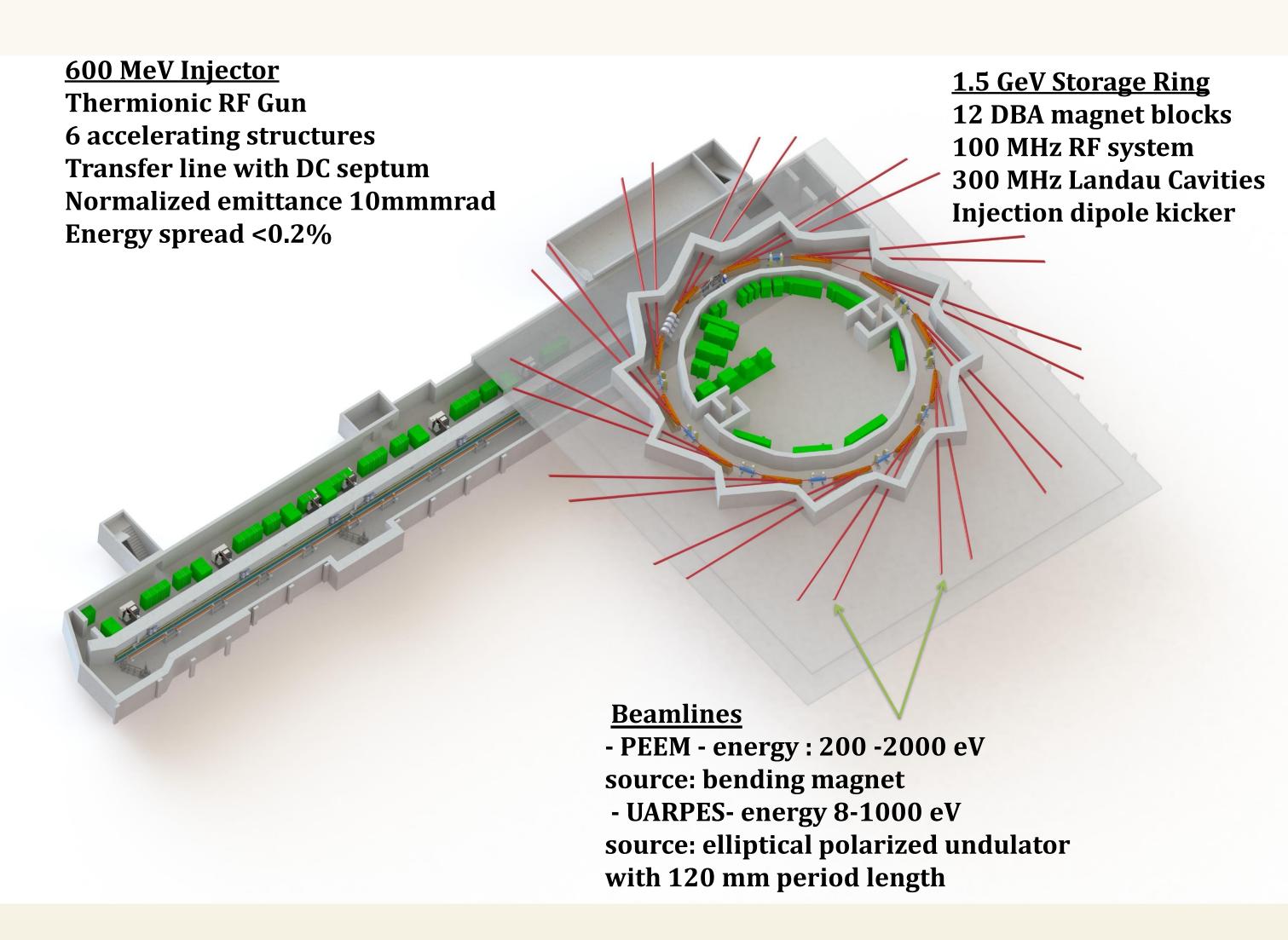
### ABSTRACT

A NSRC SOLARIS is a synchrotron light source in Krakow, Poland. The control systems for the linac, storage ring and beamlines are fully operational and are used in the commissioning process.

#### **SOLARIS OVERVIEW**

## **TANGO CONTROL SYSTEM**

Software platform for the control system is Tango Controls. The control system based on Tango Controls has a lot of elements: Tango Host server with database, archiving system, high level and low level software. At Solaris there are three instances of Tango: one for a linac and a storage ring (Tango 9) and one per each of two beamlines (Tango 8 at UARPES and TANGO 9 at PEEM/XAS). They are responsible for acquisition of more than 5000 signals. An archiving system is using TDB and HDB tools from Soleil. At PEEM/XAS beamline are there are held tests of HDB++ archiving system. At Solaris low level applications are developed in the Python programming language using an API to the Tango core - the PyTango package. Device servers are used for connection of hardware to the control system. The facade device library from MAX IV (Lund, Sweden) is used for high-level Tango devices. The Taurus package from ALBA (Barcelona, Spain) is used for writing a high level software, like GUIs. For browsing Tango database and checking each device operators use an open source application ControlProgram. The ControlProgram is also used for running Tango tools and another GUIs.



NView Search PLC/IMAG/BOOLEAN	Int Test Device #1 PLC/IVAC/BOOLEAN	erlock GUI 2	
PLC/IVAC/BOOLEAN RUNNING			E Device Panel
Bypass interlock of IPCU3 from TANGO	No description	Vacuum alarm from IPCU1 to TANGO	Valve status closed to TANGO
B_I_TLCAB02_VAC_IPCU2_Bypass_C	B_I_TLCAB02_VAC_IPCU7_Bypass_C	B_I_S00_VAC_VGMB1_Interlock_S	B_I_S01B_VAC_VGMB1_Open_C
Bypass interlock of IPCU2 from TANGO	Bypass interlock of IPCU7 from TANGO	Valve Interlock status to TANGO	Open valve from TANGO
B_I_TR1_VAC_VGMB2_Open_S	B_I_TR1_VAC_VGMB1_Interlock_S	B_I_S03B_DIA_SCRN1_Insert_C	B_I_K02CAB05_VAC_IPCU4_Bypass_C
Valve status open to TANGO	Valve Interlock status to TANGO	Insert YAG from TANGO	Bypass interlock of IPCU4 from TANGO
B_I_K01CAB05_VAC_IPCU6_Bypass_C	B_I_S01B_VAC_VGMB1_Open_S	B_I_TR1_DIA_SCRN1_Insert_C	B_I_K00CAB01_VAC_IPCU6_A
Bypass interlock of IPCU6 from TANGO	Valve status open to TANGO	Insert YAG from TANGO	Vacuum alarm from IPCU6 to TANGO
B_I_TR1_DIA_SCRN2_Insert_C	B_I_K02CAB05_VAC_IPCU5_Bypass_C	B_I_K00CAB01_VAC_IPCU5_Bypass_C	B_I_TR1_VAC_VGMB1_Error_S
Insert YAG from TANGO	Bypass interlock of IPCU5 from TANGO	Bypass interlock of IPCU5 from TANGO	Valve Error status to TANGO
B_I_TR1_DIA_SCRN2_Out_S	B_I_K03CAB05_VAC_IPCU4_Bypass_C	B_I_TR1_VAC_VGMB1_Close_C	B_I_S01B_VAC_VGMB1_Closed_S
YAG status extracted to TANGO	Bypass interlock of IPCU4 from TANGO	Close valve from TANGO	Valve status closed to TANGO
B_I_K03CAB05_VAC_IPCU8_A	B_I_TR1_DIA_SCRN1_Out_S	B_I_S02B_DIA_SCRN1_Extract_C	B_I_K02CAB05_VAC_IPCU8_A
Vacuum alarm from IPCU8 to TANGO	YAG status extracted to TANGO	Extract YAG from TANGO	Vacuum alarm from IPCU8 to TANGO
B_I_K00CAB01_VAC_IPCU1_Bypass_C	B_I_TR1_VAC_VGMB2_Closed_S	B_Reset_C	B_I_K01CAB05_VAC_IPCU6_A
Bypass interlock of IPCU1 from TANGO	Valve status closed to TANGO	Reset all errors/alarms from TANGO	Vacuum alarm from IPCU6 to TANGO
B_I_K00CAB01_VAC_IPCU7_Bypass_C	B_I_TLCAB02_VAC_IPCU1_Bypass_C	B_I_S01B_VAC_VGMB1_Close_C	B_I_K01CAB05_VAC_IPCU8_Bypass_C
Bypass interlock of IPCU7 from TANGO	Bypass interlock of IPCU1 from TANGO	Close valve from TANGO	Bypass interlock of IPCU8 from TANGO
B_I_TLCAB02_VAC_IPCU2_A	B_I_S01A_DIA_SCRN1_Out_S	B_I_TR1_VAC_VGMB2_Close_C	B_I_K00CAB01_VAC_IPCU3_Bypass_C
Vacuum alarm from IPCU2 to TANGO	YAG status extracted to TANGO	Close valve from TANGO	Bypass interlock of IPCU3 from TANGO
B_I_K01CAB05_VAC_IPCU4_Bypass_C	B_I_K01CAB05_VAC_IPCU4_A	B_I_S01A_DIA_SCRN1_Extract_C	B_I_TR1_VAC_VGMB2_Error_S
Bypass interlock of IPCU4 from TANGO	Vacuum alarm from IPCU4 to TANGO	Extract YAG from TANGO	Valve Error status to TANGO
B_I_K01CAB05_VAC_IPCU2_A	B_I_DIA_BPM_INTERLOCK_A	B_I_K02CAB05_VAC_IPCU8_Bypass_C	B_I_K02CAB05_CTL_K01_A
Vacuum alarm from IPCU2 to TANGO	Global alarm from all injector's BPMs	Bypass interlock of IPCU8 from TANGO	Indicates problems with PLC remote I/Os
	Loading done.		Reload 🚫 Abo

Fig. 1 SOLARIS layout

Fig. 3 Panel for displaying alarms from PLCs

## **CONTROL SYSTEM ELEMENTS**

Control system for the linac, storage ring and each beamline consists of the following parts:

- Instance of the Tango DB:
  - Applications facilitating access to the devices' parameters
  - Diagnostic & management tools
- Archiving:
- Historic Database •
- Temporary Database
- Sardana (at beamlines):
  - Configuration of measurements synchronised with motorisation
- Instance of SOLARIS Control Program and related custom GUIs
- Hardware.

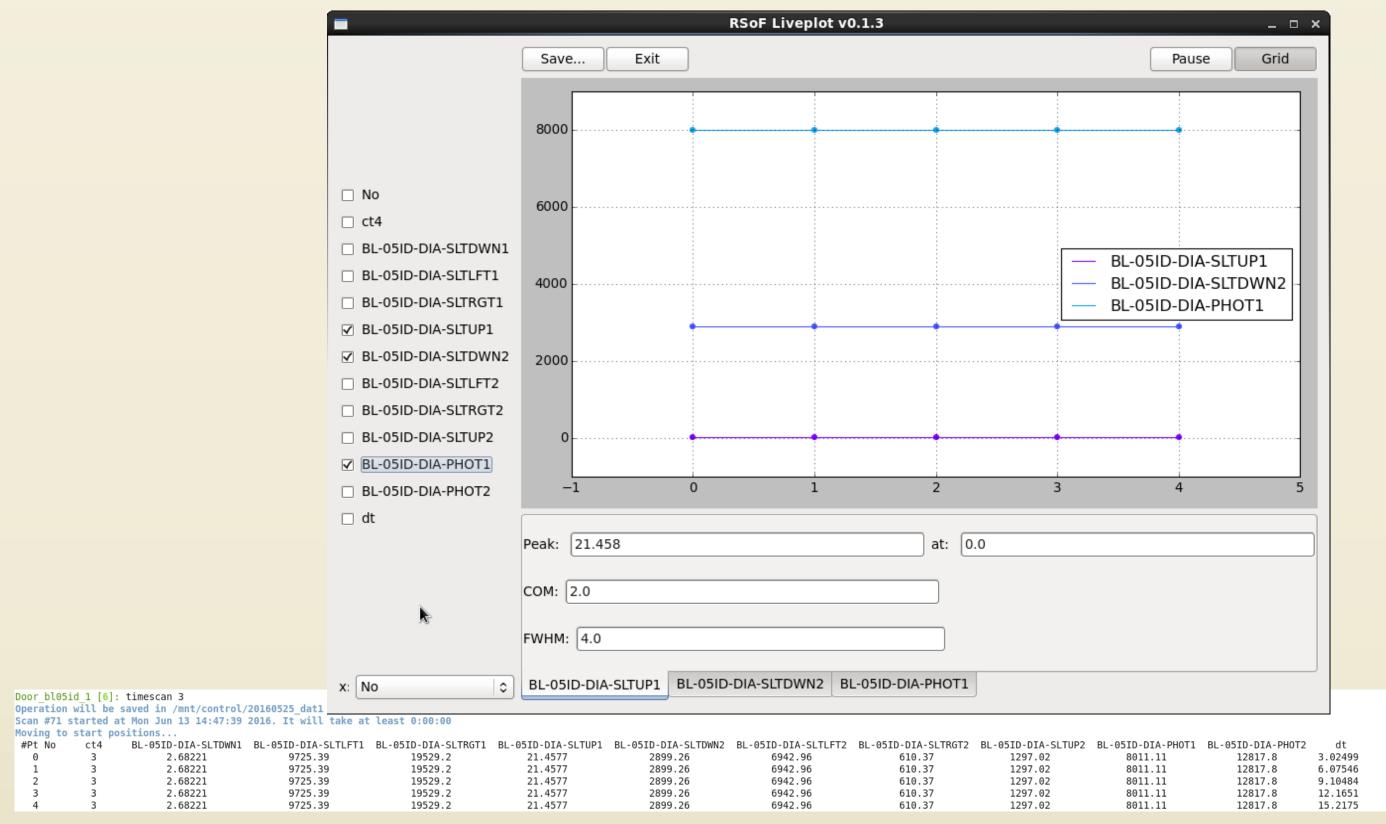
All signals are accessible from workstations near beamlines and from the Control Room.

Solaris Synchrotron Control Pro	]		
File Edit Tools			
Device Filtering			
Section: 🖌 Subsystem: 🖌 Class:	✓ X		
Expand Collapse	Select All		
Device Tree Device List Device Groups		VAC_R1_07	0
Device		ABSORBERS	R1-07-VAC-VGMB1
E I-K00			
E HOT E MAG		Flow Alarm 1:  Chow Alarm 1:  Chow Alarm 3:  Chow A	
		lescription available	
I-K00/MAG/I-K00CAB02-MAG-PS0			UNKNOWN
	ION PUMPS ACHROMAT CHAMBER		

# SARDANA, SCANS AND SPOCK

Sardana is an open-source framework serving as SCADA (Supervisory, Control And Data Acquisition). Its main role is to provide interface for performing scans (continuous movements of motors and synchronised acquisition from various sources) which are essential in conducting experiments. The results can be plotted live and stored for later processing. It is also integrated with Tango and facilitates communication with motorisation.

There are two ways of using Sardana: a command line interface (CLI) or graphical user interfaces (GUIs). The former is called Spock – it imitates SPEC from ESRF, Grenoble in functionality, but it's based on Python's interactive console, IPython. The latter is built using Taurus – PyQt-based GUI library for Tango. Operators can create simple panels with just clicking and programmers can benefit from the application programming interface (API) in preparing more advanced GUIs.



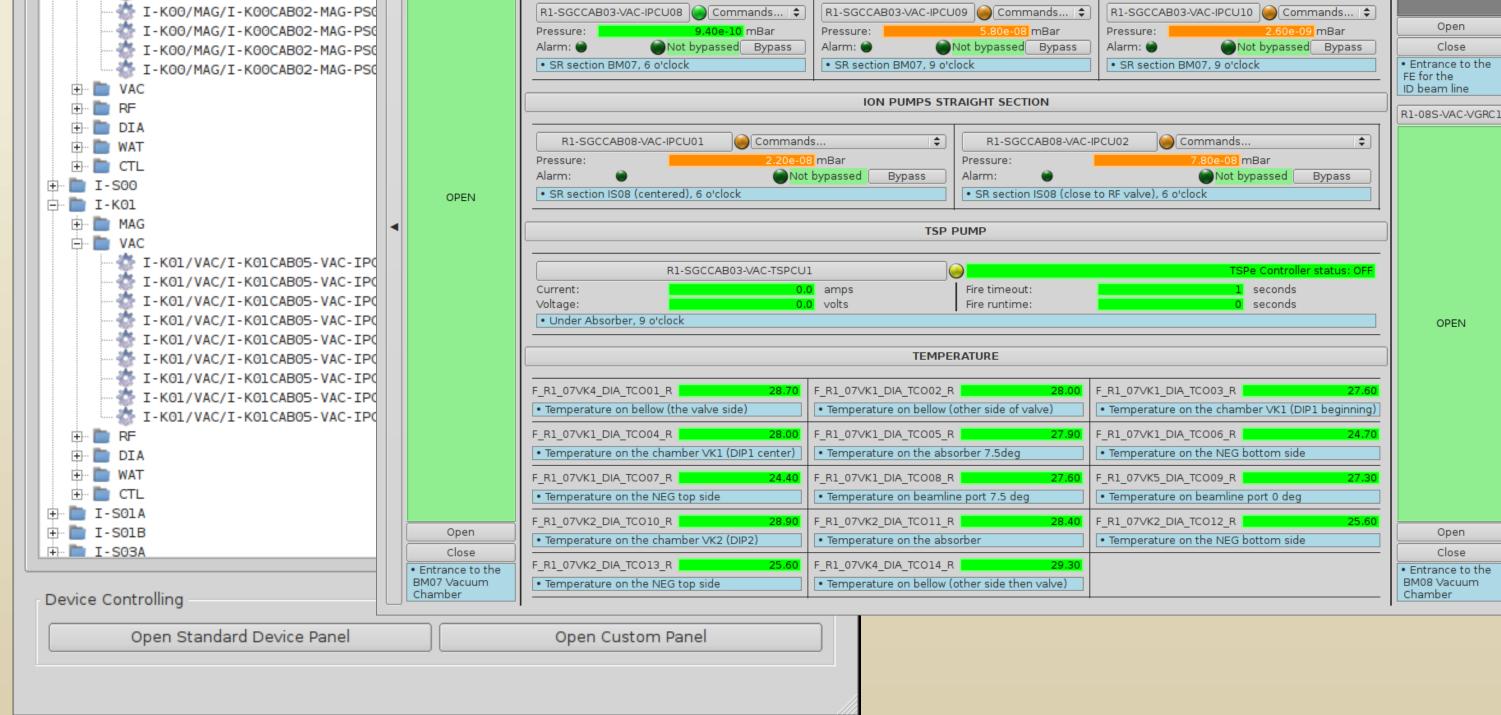


Fig. 2 SOLARIS Synchrotron Control Program

Fig. 4 Scanning in Sardana's CLI, Spock, and live plotting

#### ACKNOWLEDGMENTS

Special thanks to Max IV, Elettra, ESRF, SOLEIL and ALBA synchrotrons colleagues who has shared experience and helped extensively during construction of Solaris.

#### REFERENCES

[1] A.I. Wawrzyniak et al., "SOLARIS storage ring commissioning", IPAC'16, Busan, Korea, WEPOW029. [2] G. Abeille et al., "Tango Archiving Service Status", ICALEPCS'11, Grenoble, 2011 [3] Ł.J. Dudek et al., "Managing beamlines at SOLARIS from an IT point of view", ISSRNS'16, Ustroń, Poland [4] Tango Controls website: <u>http://www.tango-controls.org</u> [5] W.T. Kitka et al., "Control System at SOLARIS synchrotron", ISSRNS'16, Ustroń, Poland