

A large, intricate wireframe model of a roller coaster track dominates the background. The track is composed of many interconnected wireframe segments, forming a complex loop that surrounds the central text. It features several curves, drops, and a small section that resembles a station or a cluster of buildings.

Farewell LabVIEW

It's been a good time!

EKS, H.Brand@gsi.de

- Modifizierte Version des Vortrags am 6. September 2022 <https://indico.gsi.de/event/15599/>
- Motivation
- Status & Zukunft
- History
 - CS
 - NI Actor Framework
 - **CS++**
 - Publication: FRS-IC
 - NI Tools
- Migration
 - Python / pykka / PyAcdaq / paho-mqtt
 - Mosquitto / MQTT-Explorer / Telegraf / InfluxDB

- *Following is my personal impression from the past ~5 years.*
- NI changed the business model
 - Focus on Automated Test Systems
 - Mobile Communication, Automotive, Semiconductors
 - Out of focus
 - Academics & Research
 - Service oriented
 - **Subscription based licensing!**
 - Packaging of open source to products, e.g. SystemLink
 - Reduced quality of support (*reported also by others*)
 - Reduced communication with customers (*personal experience*)
 - *No contact from NI since the Enterprise Agreement was established.*
 - **No information about security notifications to GSI since years!**
 - Hardware sales by 3rd-Party distributors with bad consulting
 - Bad experience with Software Platform Bundle 2021
 - LabVIEW 2021 / SP1 / f2, RT, DSM are buggy, compared to LV2020.
 - Bug-fixes announced for LV2022Q3, but not all of them are really fixing the problems.
 - Sudden deprecation of products, e.g. <https://www.ni.com/pdf/manuals/lv-rt-2022-q3.html>
 - *LabVIEW 2022 Q3 Real-Time Module 64-bit does not support Modbus IO server. As a workaround, you can use the option of direct communication using the Modbus LabVIEW API.*
 - *LabVIEW 2022 Q3 Real-Time Module 32-bit and 64-bit no longer supports Real-Time Trace Viewer. You can **use KernelShark, a similar open source tool, as an alternative.***
 - *No insight into LabVIEW Scheduler*

- NI switched to a subscription model in 2022
 - Start prices are reduced, but
 - Increased long-term license cost compared to SSP
 - **Loosing access to LabVIEW if not continued.**
 - **There is no free LabVIEW Viewer!**
 - **Community Edition does not support all Modules.**
- Enterprise Agreement until 14. December 2025
 - And then?
 - Fallback for GSI/FAIR/HIM
 - Perpetual Licenses
 - 39 x LabVIEW Full-Development
 - 9 x Control Test Plus
 - Some Deployment Licenses
 - Application Builder
 - Data Logging and Supervisory Control Module
 - IMAQdx
 - DIAdem?
 - IMHO: We are able to freeze any time.
 - We, HGF, should start a serious discussion soon!

Status Quo - Vertrag



NI Enterprise Agreement

- HGF Rahmenvertrag wurde unter der Federführung von DESY verhandelt.
- Offizieller Vertragspartner von NI für die HGF ist stellvertretend das DLR.
- In Kraft seit 15.12.2020, Laufzeit 5 Jahre mit festgelegten jährlichen Lizenzgebühren
- <https://wiki.gsi.de/NIUser/EnterpriseAgreement>

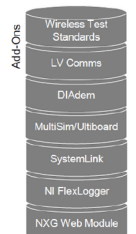
Umfang

- Unlimitierter Zugriff auf *Control Test Plus Suite*
- Add-Ons, Deploy- und Debug-* Lizenzen müssen zusätzlich über das DLR separat erworben werden. (möglichst im November)
- GSI betreibt einen Volumen-Lizenz-Manager für GSI/FAIR und HIM
 - HIJ hat einen Vertrag mit Uni Jena
 - Vierteljährliche Nutzungsberichte an NI, ohne Rückmeldung der relativen Anteile in der HGF

Enterprise Agreement offers **Unlimited Term Seat** usage from the following Suite bundles:

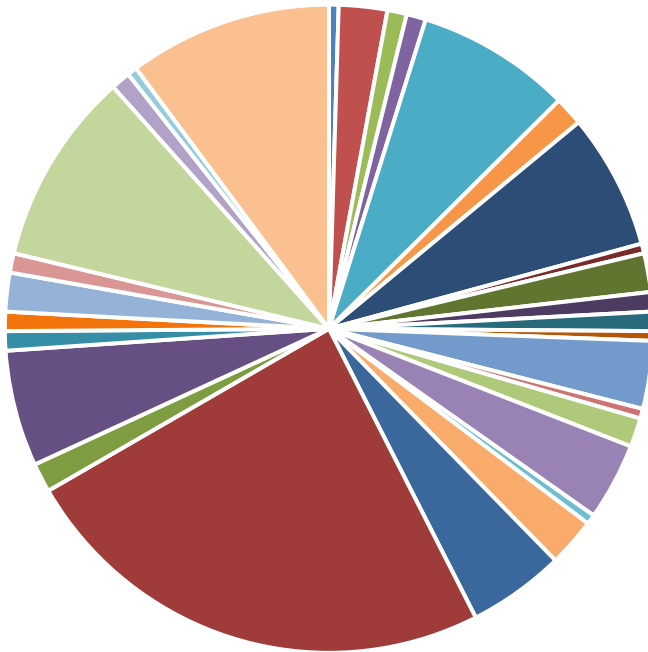
	Core	Test	Control	Control Test
Standard	LabVIEW (Win/Mac/Linux) LabVIEW NXG LabWindows/CVI Measurement Studio Signal Express Statechart Module Mathscript RT Module Automotive Diagnostics Command Set ECU Measurement & Calibration Electrical Power Suite Industrial Comm for Ethernet/IP Digital Filter Toolkit Data Finder Toolkit	ADD to Core software: TestStand Switch Executive Requirements Gateway ATML Toolkit	ADD to Core software: Real-Time Module FPGA Module	ADD to Core software: Real-Time Module FPGA Module TestStand Switch Executive Requirements Gateway ATML Toolkit
Plus	ADD to Standard: Modulation Sound & Vibration Vision Module Vision Builder AI Motion (Module & Assistant) Lego Mindstorms Module	ADD to Standard: Modulation Sound & Vibration Vision Module Vision Builder AI Motion (Module & Assistant) Lego Mindstorms Module	ADD to Standard: NI Veristand Control & Design Simulation DSC Module Modulation Sound & Vibration Vision Module Vision Builder AI Motion (Module & Assistant) Requirements Gateway Lego Mindstorms Module	ADD to Standard: NI Veristand Control & Design Simulation DSC Module Modulation Sound & Vibration Vision Module Vision Builder AI Motion (Module & Assistant) Lego Mindstorms Module

Add-Ons are available as limited seats only



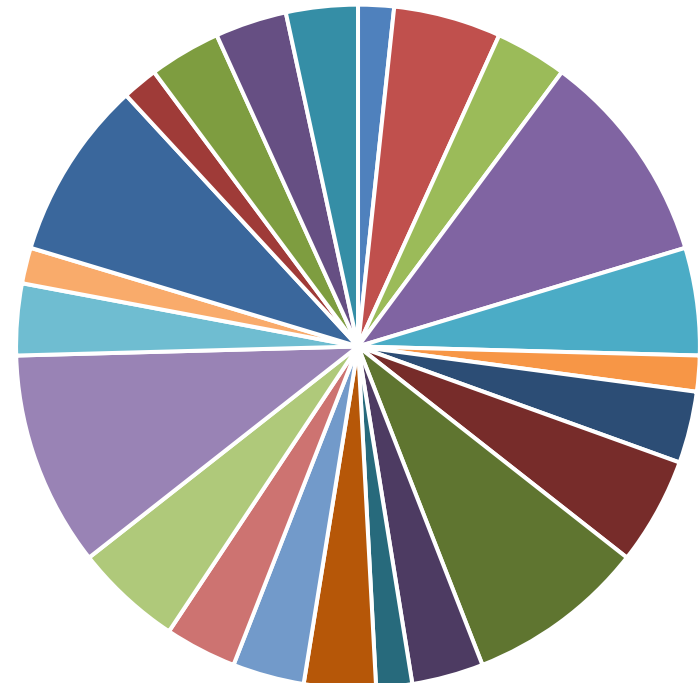
- **Benutzer basierte Lizenzen (59 am 26.08.2022)**
 - Ein Entwickler kann die SW auf beliebig vielen Rechnern installieren, aber nur auf einem einzigen Rechner gleichzeitig benutzen.
 - Der Volumenlizenzmanager lässt aber eine gleichzeitige Nutzung auf verschiedenen Rechnern zu, solange die maximale Anzahl noch nicht ausgeschöpft ist, derzeit 1000.
 - Ob das formal juristisch in Ordnung ist, kann ich nicht beantworten.
 - Wie die gleichzeitige Nutzung in die Bewertung für die Zukunft eingeht, weiß ich auch nicht.
- **Computer basierte Lizenzen (207 am 26.08.2022)**
 - Beliebig viele Benutzer können die SW auf einem Computer benutzen.
 - **Disconnect Lizenzen (9 von 207 am 26.08.2022)**
 - können weiterhin gewährt werden.

Anteile / Abteilung
Computer basiert



1	2	3	4	5	6	7	8	9	10
11	12	13	14	15	16	17	18	19	20
21	22	23	24	25	26	27	28	29	30

Anteile / Abteilung
Benutzer basiert



1	2	3	4	5	6	7	8	9	10	11	12
13	14	15	16	17	18	19	20	21	22	23	

- Deutliche Preissteigerung beim Übergang von ICLA zum EA in 2020
- HGF Rahmenvertrag gilt für 5 Jahre
 - Bei Ausstieg eines HGF Zentrums werden die Kosten auf die verbleibenden aufgeteilt.
 - Das könnte zu einem Ausstiegsszenario mit Dominoeffekt bei Anschlussverhandlungen führen.
 - Welche Preissteigerungen sind zu erwarten?
 - Inflationsausgleich
 - Orientierung an der Nutzung der Software im aktuellen Vertragszeitraum
 - Es ist ein vierteljährlicher Bericht an NI mit den Nutzungsdaten vereinbart worden.
 - Mir ist unbekannt, wie daraus ein Preis für einen Nachfolgevertragszeitraum abgeleitet werden soll.
 - Kostensenkung erscheint aber unwahrscheinlich.
 - Kommt kein Anschlussvertrag zustande, werden **Listenpreise** fällig!
 - Empfehlung des Koordinierungsausschuss für Datenverarbeitung
 - „Das Überschreiten der Lizenzzahl oder besser formuliert der Zuwachs an genutzten Lizenzen ist für die Laufzeit zunächst egal und gewollt, **aber am Ende der Laufzeit sollte es aber keinen Megazuwachs gegeben haben**, sonst wird die neue Periode der ULA ab dem 6. Jahr ggf. entsprechend teurer!“
 - „Generell ist die HGF-Zentren Strategie möglichst die Nutzung von LabView zu reduzieren, das sollten wir für die Zukunft berücksichtigen um unangenehme Überraschungen nach Ablauf eines evtl. jetzt abgeschlossenen Rahmenvertrags zu vermeiden.“
- Risiken durch Änderungen des Geschäftsmodells von NI
 - NI warb früher damit, dass kein Marktsegment mehr als 10% des Umsatzes ausmacht.
 - NI reduzierte auf 6 Geschäftsfelder
 - Halbleitertechnik, Automobil, Luft- und Raumfahrt sowie Verteidigung
 - Elektronik, Energie, Forschung und Lehre
 - Heute sind es nur noch 4



- Emerson to Advance Global Automation Leadership Through Acquisition of NI
<https://www.emerson.com/en-us/news/2023/04-emerson-ni-acquisition>

- Wie kann das finanzielle Risiko begrenzt werden?
 - Nicht limitierter Zugriff
 - Beobachtung der Nutzung und Warnung an die GF bei Überschreiten ein Limits?
 - Limitierter Zugriff
 - Genehmigungsprozess würde notwendig werden. Beispiel: MS Office in der IT
 - Budgetierung zentral (EEL) oder durch die Abteilungen bzw. Kostenstellenverantwortliche
- Wie kann ein Vendor-Lock vermieden werden?
 - Benutzen offener Netzwerk-Schnittstellen für den Fernzugriff und offene Datenformate
 - Erstellen von Executables
 - Deployment-Lizenzen fallen nur einmalig bei der Beschaffung an.
 - Damit könnte man NI SW einfrieren und mit den bereits erworbenen Lizenzen eine Weile überleben.
- Welche Alternativen gibt es?
 - Bei GSI/FAIR:
 - EPICS, Python, PyAcdaq/Pykka, Tkinter/QT, FESA/Java, Siemens TIA, Beckhoff/TwinCat, Pures C/C++
 - NI LV-DSC -> EPICS (ALH/Archiver); InfluxDB
 - NI-Vision -> OpenCV; Other stuff: Open Source Libraries
 - NI-RT wird als RT praktisch nicht benutzt. NI-FPGA: Expertise für FPGA Programmierung gibt es bei GSI.
 - Community: DOOCS, TANGO, TINE

Future Demand for LabVIEW@GSI/FAIR

We are save with fallback licenses!



Abteilung	Support	FDS W	FDS L	App-Builder	DSC	DSC-Deploy	FPGA	Vision	Vision-Deploy	IMAQdx	Control-Test-Plus
Sporadisch											
Targetlabor	EKS										
ENG			3								
SHC											1
PHELIX					1						
DTL (BV)		1			1						
DTL (MT)		1			1						
ATP (TL)		1									
DEC											1
RRF	EKS										1
Total			6	0	3	0	0	0	0	0	3
Permanent											
EKS											2
VAC		1			1						
ENG		3					1				
PHELIX		10			1				4	65	
MAT											1
DTL(BV)		1			1			2			2
DTL (MT)		2			1			1			2
ATP (TL)		3									
SHE-P		14			1			2			3
DEC	EKS	1				1	1				
Total			35	0	5	1	2	5	4	65	9

Status on 13.04.2023

- Einschränkungen bei der Wartbarkeit vorhandener Systeme
 - NI Hardware kann weiterverwendet werden, weil sie mit ANSI-C-Treibern ausgeliefert werden und teilweise Python Bibliotheken zur Verfügung stehen.
- Technologiebrüche
- Neu-Implementierung von Systemen
- Aufbau von Expertise mit alternativen Systemen
- Schulung für das Arbeiten und Entwickeln mit den neuen Systemen
- Einsparungen bei Lizenzkosten
 - Bei Einsatz von Open Source Lösungen
- Rolle der Kontrollsystemgruppe der EEL
- Probleme in der HGF (*Sippenhaft*)

- My first App@GSI
 - TargetScanner@TargetLab
 - Pure LabVIEW 5.0 Prof.Dev. in 1999 + Database & Report Generation
- Extended support at GSI
 - LabVIEW Instrument Drivers and Libraries
 - Control System Applications
 - Supervision of students as well as Diplom- and Bachelor-Thesis
 - FPGA
 - Multi Channel Scaler & Pulse Pattern Generator
 - TASCA Interlock & Current Measurement
 - NI-Scope for VITO, CERN
- CS Framework in 2001
 - V1 based on ObjectVIEW
 - V2 own Core, Network: Queues and TCP/IP
 - V3 Network: DIM
 - Special thanks to D. Beck!
- LVOOP in 8.5
 - Originally in 8.2, but dynamic dispatching was not reentrant
- NI Actor Framework in LabVIEW 2012
- **CS++** in 2014 @STF
 - NIM-A: A. State et.al. 2022 *The slow control system of the FRS Ion Catcher*

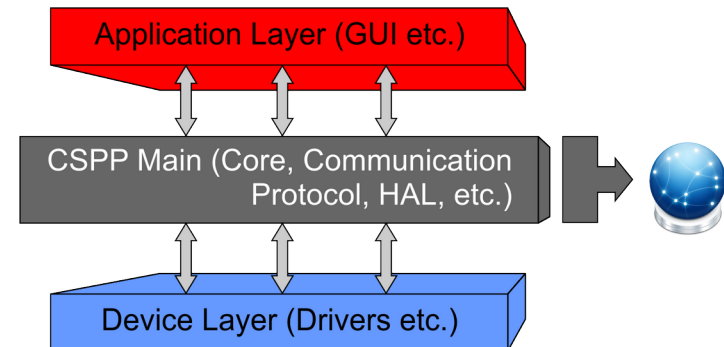
- CS Framework
 - CS is a LabVIEW/DIM based control system framework
 - multi-threaded, event driven, object oriented and
 - distributed with SCADA functionality.
 - An experiment control system can be developed by combining the CS framework with experiment specific add-ons.
 - CS is supported on MS-Windows and on Linux (real-time OS Pharlap, LabVIEW RT)
- Artificial object-oriented approach started with LabVIEW 6i
 - Reference based (VI-Server), Multiple Inheritance like C++
 - *Complex with many recommendations which cannot be enforced.*
- Network layer: **D**istributed **I**nformation **M**anagement (DIM)
- Mainly used with Laser (PHELIX, POLARIS), many Iontraps FOPI, RISING, CaveA MotionControl

- Successful Feasibility Study: Mobile Agent
 - Based on LVOOP (LabVIEW 8.5)
 - Diploma Thesis
- NI Actor Framework provides simple and efficient design
 - Released with LV 2012
 - First application: Gas Flow Control for the COMPACT Detector @TASCA
- Profit from NI maintenance and community developments
- Integrate non LabVIEW experts like short term Bachelor & Master students
- CLAD level implementing derived classes using a cooking recipe
- Use as much NI Tools as possible
 - Data Logging & Supervisory Control Module (DSC)
 - Distributed System Manager (DSM)
 - TDMS & DIAdem
 - (Teststand)

Migration I

CS Framework → CS++

- **Similar feature set** as CS Framework based on
 - LVOOP & Dataflow
 - Actor Framework
- **Plan:** Extent Actor Framework
 - Concentration on necessary features for our Control Systems!
- **Architecture:** 3-Layer



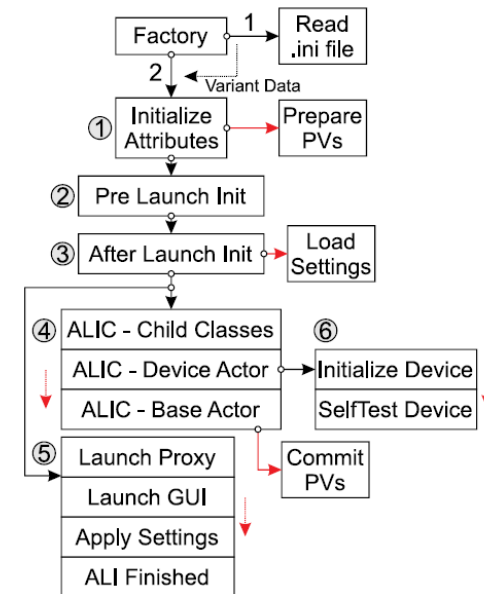
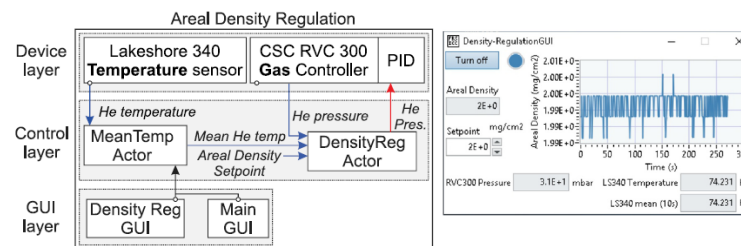
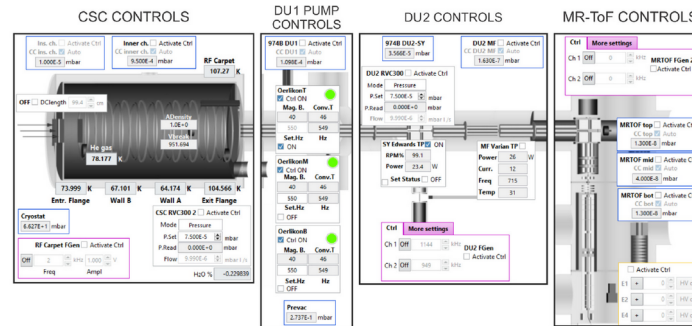
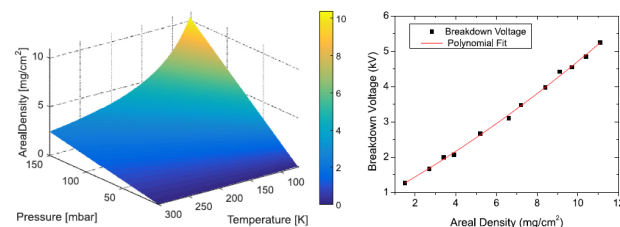
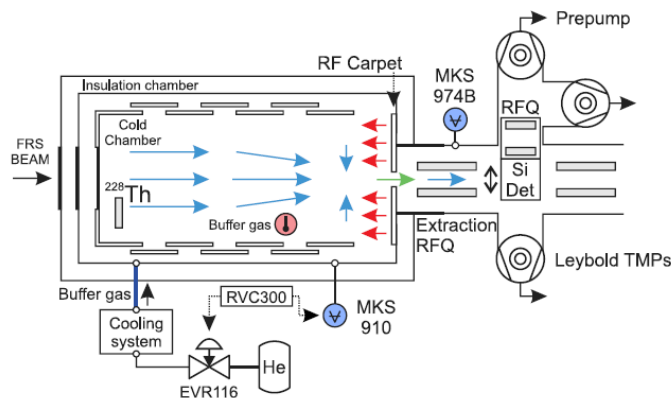
- **Developer groups**
 - Core Developer:
 - **CS++** core and add-on libraries
 - Profound LVOOP/AF and **CS++** knowledge
 - Local Developer:
 - experiment-specific GUI- and Device-Actors
 - ~CLAD knowledge sufficient
 - User:
 - Use and/or configuration of Application

The slow control system of the FRS Ion Catcher

- A. State et.al. Nuclear Inst. and Methods in Physics Research, A 1034 (2022) 166772
<https://doi.org/10.1016/j.nima.2022.166772>

Abstract

The **FRS Ion Catcher** setup at GSI is used for **high precision measurements of slowed down exotic nuclei**. The current setup consists of a gas-filled stopping cell, working at cryogenic temperatures for improved gas purity, coupled to a radio-frequency quadrupole beamline for bunching and pre-selection and a multiple-reflection time-of-flight mass spectrometer for precise mass measurements. The setup with all its components is controlled by various control systems. The development of a new slow control system for monitoring, control, and logging of all components of this setup is presented. The slow control is based on the **LabVIEW Actor Framework enhanced by the Control System++ libraries**, a highly scalable and extendable platform which is a powerful tool to create complex messaging schemes from custom events. The implementation and testing of the control system at the FRS Ion Catcher setup is discussed.



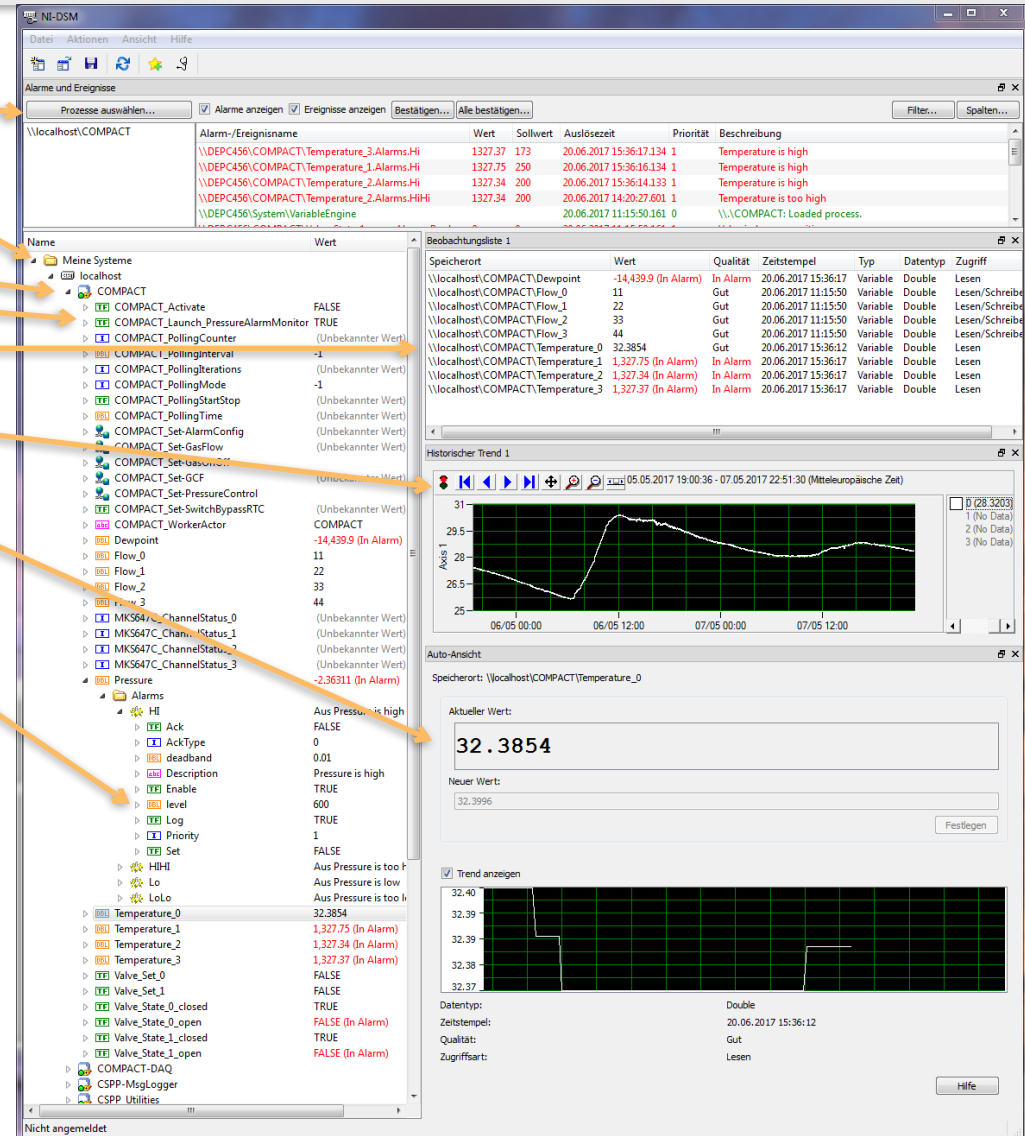
*ALIC - After Launch Init Core

- Measurement & Automation Explorer
 - Necessary and useful for NI hardware configuration and test.
- LabVIEW
 - Data Logging and Supervisory Control
 - Real Time
 - FPGA
- Shared Variables
- Distributed System Manager
- DIAdem

- Built-in networked database for distributed data logging
- Graphical development for HMI and distributed monitoring and control applications
- Configuration-based real-time and historical trending, alarms, and event handling
- Classic OPC client and Modbus for data sharing with third-party devices
- Unlimited tags to connect to NI hardware, PLCs, and other third-party hardware
- Deployment requires a LabVIEW DSC Module Run-Time System license

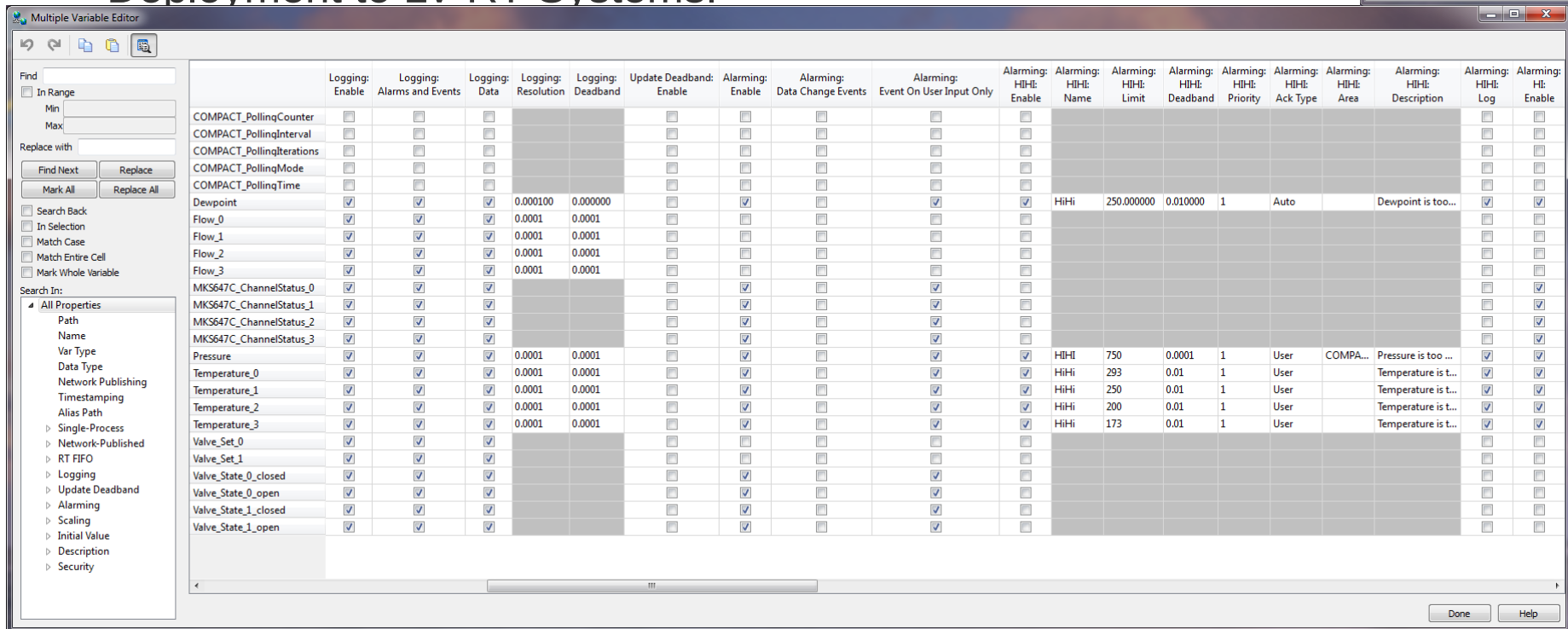
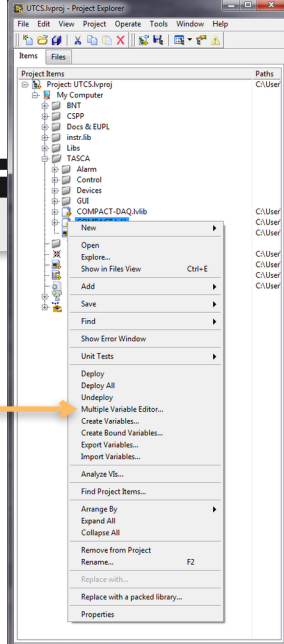
Distributed System Manager

- Alarm & Event Monitoring
- Distributed Process List
- Process
- Shared Variable List
- Selected SV List
- Historical Data Trending
- Online Monitor/Trending
- Processes, (Bound-)SV, Datalogging and Alarming can be configured interactively from DSM.
- Sections can be un-/docked
- Very useful for adhoc monitoring and debugging without explicit GUI programming.



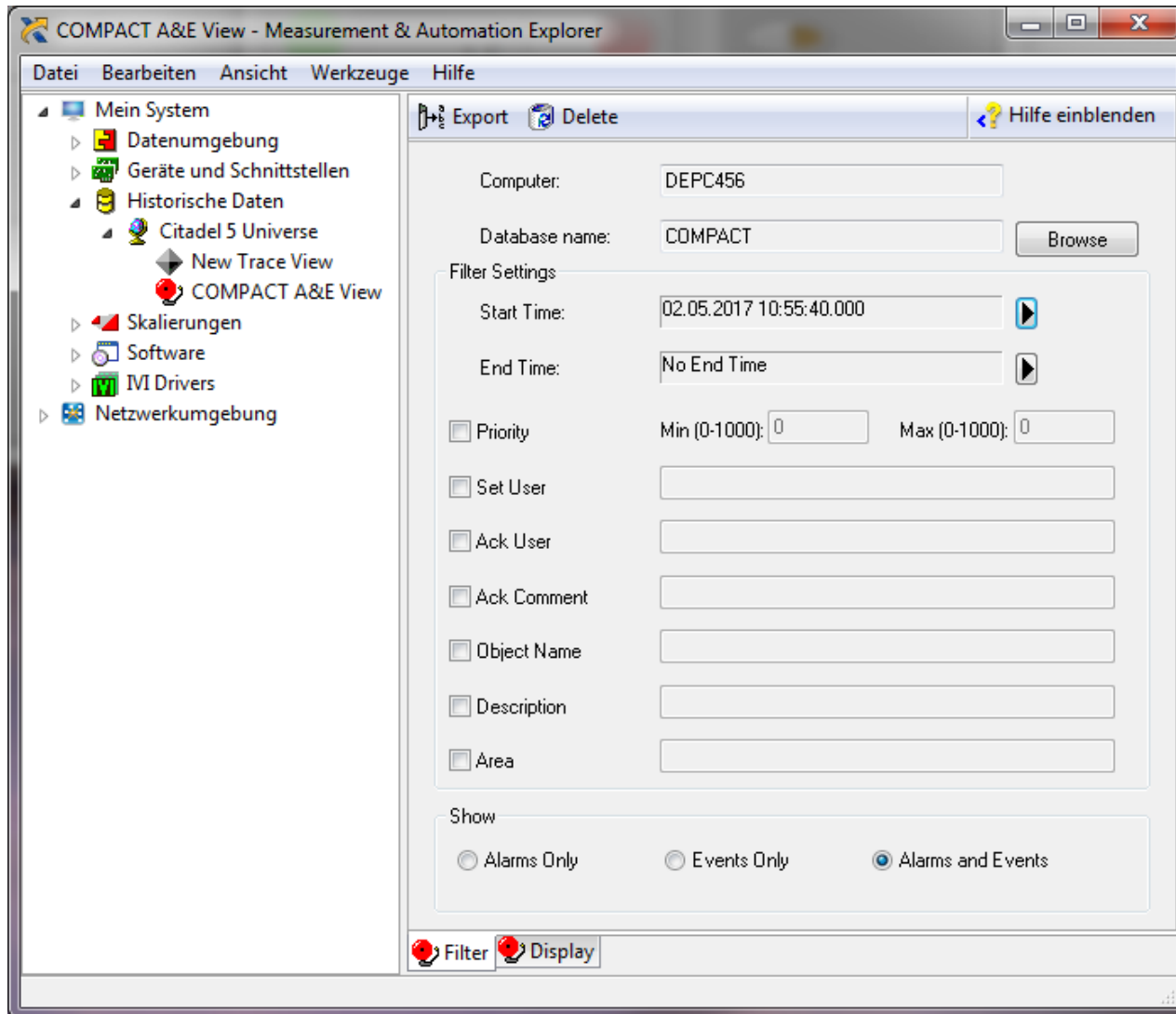
LabVIEW SV.lvlib

- Same configuration options as DSM plus:
- Mass configure Shared Variables Libraries using the Multiple Variable Editor in LabVIEW.
- Export/Import Shared Variables using CSV-Format.
- Deployment to LV-RT-Systems.

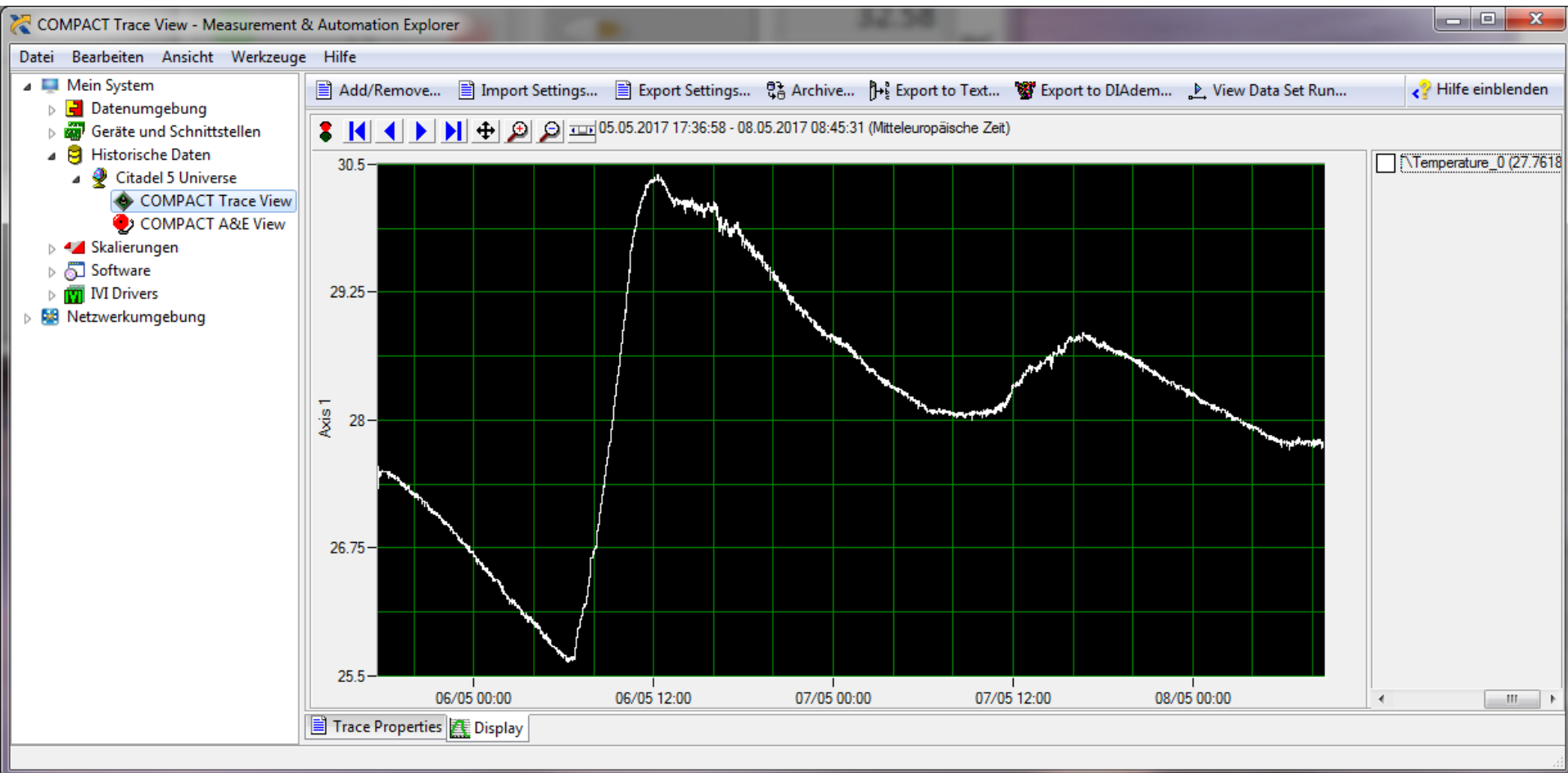


Measurement & Automation Explorer

Distributed Alarm & Event Filter



Measurement & Automation Explorer Historical Trend Viewer



Measurement & Automation Explorer Distributed Alarm & Event View



COMPACT A&E View - Measurement & Automation Explorer

Datei Bearbeiten Ansicht Werkzeuge Hilfe

Mein System
 ▶ Datenumgebung
 ▶ Geräte und Schnittstellen
 ▶ Historische Daten
 ▶ Citadel 5 Universe
 ▶ New Trace View
 ▶ COMPACT A&E View
 ▶ Skalierungen
 ▶ Software
 ▶ I/O Drivers
 ▶ Netzwerkumgebung

Export Delete
 Refresh (Filter Settings Have Changed) Hilfe einblenden

Object Name	Set Time	Set User	Ack Time	Ack User	Clear Time	Clear User	Priority	Area	Value	Setpoint	Description	Ack Comment
\\DEPC456\CO...	20.06.2017 15:41:24.141	(Nobody)					1	COMPACT	1327.370974	173.000000	Temperature is high	
\\DEPC456\CO...	20.06.2017 15:41:24.141	(Nobody)					1		-244.989090	10.000000	Temperature too low	
\\DEPC456\CO...	20.06.2017 15:41:24.141	(Nobody)					1		-244.989090	25.000000	Temperature is low	
\\DEPC456\CO...	20.06.2017 15:41:23.141	(Nobody)	20.06.2017 15:41:24.141	(Nobody)	20.06.2017 15:41:24.141	(Nobody)	1		-244.989090	25.000000	Temperature is low	Auto-acked
\\DEPC456\CO...	20.06.2017 15:41:23.141	(Nobody)	20.06.2017 15:41:24.141	(Nobody)	20.06.2017 15:41:24.141	(Nobody)	1		-244.989090	10.000000	Temperature too low	Auto-acked
\\DEPC456\CO...	20.06.2017 15:41:22.142	(Nobody)	20.06.2017 15:41:23.141	(Nobody)	20.06.2017 15:41:23.141	(Nobody)	1	COMPACT	1327.370974	173.000000	Temperature is high	Auto-acked
\\DEPC456\CO...	20.06.2017 15:41:21.142	(Nobody)	20.06.2017 15:41:24.141	(Nobody)	20.06.2017 15:41:24.141	(Nobody)	1	COMPACT	1327.745544	250.000000	Temperature is high	Auto-acked
\\DEPC456\CO...	20.06.2017 15:41:20.142	(Nobody)	20.06.2017 15:41:21.142	(Nobody)	20.06.2017 15:41:21.142	(Nobody)	1		-244.989090	10.000000	Temperature too low	Auto-acked
\\DEPC456\CO...	20.06.2017 15:41:20.142	(Nobody)	20.06.2017 15:41:22.142	(Nobody)	20.06.2017 15:41:22.142	(Nobody)	1		-244.989090	10.000000	Temperature too low	Auto-acked
\\DEPC456\CO...	20.06.2017 15:41:20.142	(Nobody)	20.06.2017 15:41:22.142	(Nobody)	20.06.2017 15:41:22.142	(Nobody)	1		-244.989090	25.000000	Temperature is low	Auto-acked
\\DEPC456\CO...	20.06.2017 15:41:20.142	(Nobody)	20.06.2017 15:41:21.142	(Nobody)	20.06.2017 15:41:21.142	(Nobody)	1		-244.989090	25.000000	Temperature is low	Auto-acked
\\DEPC456\CO...	20.06.2017 15:41:17.142	(Nobody)	20.06.2017 15:41:20.142	(Nobody)	20.06.2017 15:41:20.142	(Nobody)	1	COMPACT	1327.370974	173.000000	Temperature is high	Auto-acked
\\DEPC456\CO...	20.06.2017 15:41:17.142	(Nobody)	20.06.2017 15:41:20.142	(Nobody)	20.06.2017 15:41:20.142	(Nobody)	1	COMPACT	1327.745544	250.000000	Temperature is high	Auto-acked
\\DEPC456\CO...	20.06.2017 15:41:16.141	(Nobody)					1	COMPACT	1327.338734	200.000000	Temperature is high	
\\DEPC456\CO...	20.06.2017 15:41:16.141	(Nobody)	20.06.2017 15:41:17.142	(Nobody)	20.06.2017 15:41:17.142	(Nobody)	1		-244.989090	25.000000	Temperature is low	Auto-acked
\\DEPC456\CO...	20.06.2017 15:41:16.141	(Nobody)	20.06.2017 15:41:17.142	(Nobody)	20.06.2017 15:41:17.142	(Nobody)	1		-244.989090	10.000000	Temperature too low	Auto-acked
\\DEPC456\CO...	20.06.2017 15:41:16.141	(Nobody)	20.06.2017 15:41:17.142	(Nobody)	20.06.2017 15:41:17.142	(Nobody)	1		-244.989090	10.000000	Temperature too low	Auto-acked
\\DEPC456\CO...	20.06.2017 15:41:16.141	(Nobody)	20.06.2017 15:41:17.142	(Nobody)	20.06.2017 15:41:17.142	(Nobody)	1		-244.989090	25.000000	Temperature is low	Auto-acked
\\DEPC456\CO...	20.06.2017 15:41:15.141	(Nobody)	20.06.2017 15:41:16.141	(Nobody)	20.06.2017 15:41:16.141	(Nobody)	1		-244.989090	25.000000	Temperature is low	Auto-acked
\\DEPC456\CO...	20.06.2017 15:41:15.141	(Nobody)	20.06.2017 15:41:16.141	(Nobody)	20.06.2017 15:41:16.141	(Nobody)	1	COMPACT	1327.745544	250.000000	Temperature is high	Auto-acked
\\DEPC456\CO...	20.06.2017 15:41:15.141	(Nobody)	20.06.2017 15:41:16.141	(Nobody)	20.06.2017 15:41:16.141	(Nobody)	1		-244.989090	10.000000	Temperature too low	Auto-acked
\\DEPC456\CO...	20.06.2017 15:41:14.141	(Nobody)	20.06.2017 15:41:15.141	(Nobody)	20.06.2017 15:41:15.141	(Nobody)	1		-244.989090	25.000000	Temperature is low	Auto-acked
\\DEPC456\CO...	20.06.2017 15:41:14.141	(Nobody)	20.06.2017 15:41:16.141	(Nobody)	20.06.2017 15:41:16.141	(Nobody)	1	COMPACT	1327.370974	173.000000	Temperature is high	Auto-acked
\\DEPC456\CO...	20.06.2017 15:41:14.141	(Nobody)	20.06.2017 15:41:15.141	(Nobody)	20.06.2017 15:41:15.141	(Nobody)	1	COMPACT	1327.338734	200.000000	Temperature is high	Auto-acked

Filter Display

DIAdem Navigator



DIAdem - [NAVIGATOR: My DataFinder]

File Find View Settings Window Help

NAVIGATOR VIEW ANALYSIS REPORT SCRIPT

External Data

Level	Property	Value
File	<Enter a property>	= <Enter a value>

<Link search conditions with AND or OR>

My DataFinder

- Search Areas
 - LabVIEW Data
 - National Instruments
 - DIAdem 2017 (32-bit) Examples
- My Computer (*.*)
 - C:
 - D
 - Dell
 - IDE
 - Intel
 - National Instruments Downloads
 - NIFFPGA
 - Programme
 - Programme (x86)
 - Sun
 - tmp
 - User
 - Brand
 - builds
 - DSC
 - LV2016Data
 - LV2017Data
 - 2017(32-bit)
 - AppBuilderCache
 - CSPP-Settings

File Browser / Search Results

Data Portal: Internal Data

EXAMPLE

- EXAMPLE
 - Time
 - Speed
 - RPM
 - Torque
 - Noise data
 - Noise_1
 - Noise_2
 - Noise_3
 - Noise_4
 - Noise_5
 - Results_Noise data
 - StatisticsText
 - Res_Noise_1
 - Res_Noise_2
 - Res_Noise_3
 - Res_Noise_4
 - Res_Noise_5
 - Room temperatures
 - Temperature_1

Structure / List /

Base Properties

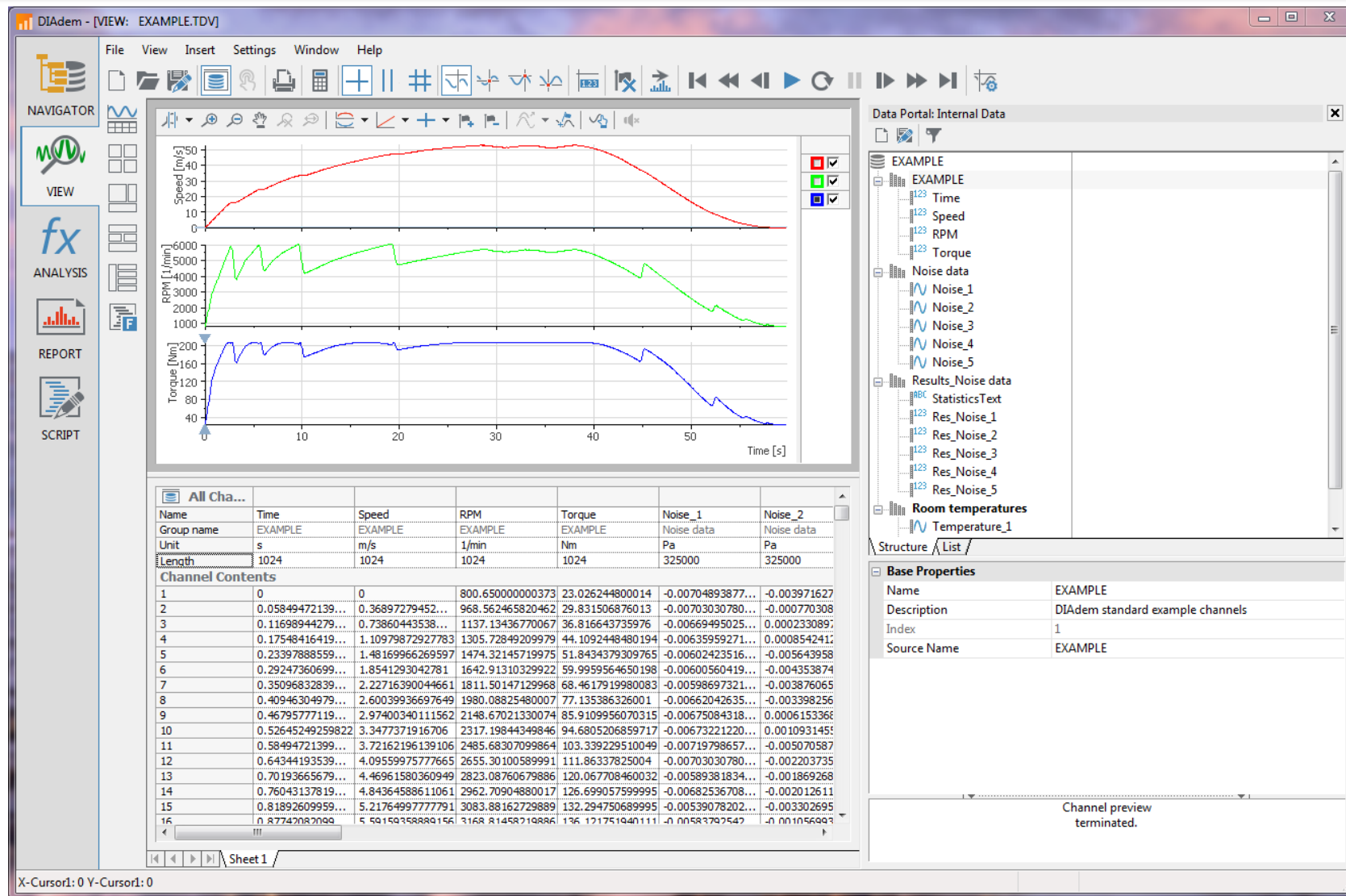
Name	EXAMPLE
Description	DIAdem standard example channels
Index	1
Source Name	EXAMPLE

Channel Preview

Automatic display is not active or the channel cannot be displayed. Use the context menu to control the display.

Channel preview terminated.

DIAdem VIEW }



- Focus of development
 - Development of hardware layer
 - Commissioning of hardware layer
 - Simple Device GUI's
 - Stability of core system
 - Device Base Classes
 - Generic interface classes
 - CS++MessageMaker
- Projects & Applications
 - Motion Control (CaveA)
 - STF
 - TASCA and COMPACT (LabVIEW-FPGA)
 - GEMDiscProduction (LabVIEW-RT-FPGA) (on-hold since 2017)
 - SHIPTRAP and HITRAP
 - FRS-IC
 - Vacuum Heating Control (LabVIEW-RT) (commissioning since 2018)
 - VITO @ISOLDE, CERN
 - More in preparation: PUMA (CERN), MLLTRAP(Orsay)
- Migration to non-LabVIEW based frameworks → **Proposal: PyAcdaq**

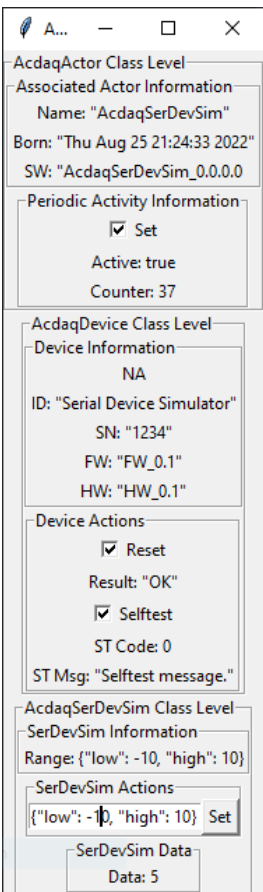
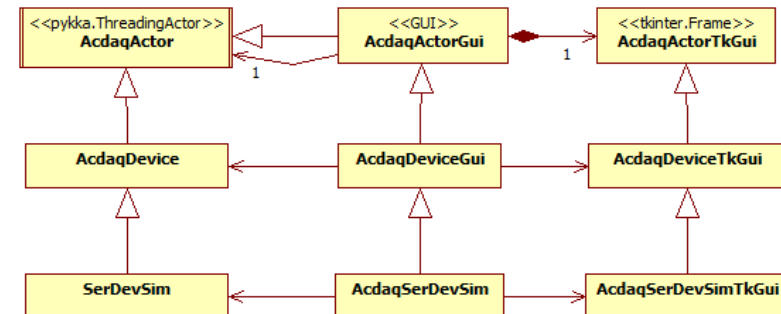
Migration Proposal → Open Source



- Measurement & Automation Explorer
 - Maybe necessary or useful for NI hardware configuration and test.
- LabVIEW → Python / C/C++ / Rust / EPICS
 - Real Time → Arduino oder RTMS
 - FPGA → VHDL
 - Shared Variables → MQTT
 - Data Logging and Supervisory Control
 - Data Logging → Telegraf / InfluxDB
 - Alarming
 - → Telegraf / InfluxDB-Task → MQTT
 - → Telegraf / Prometheus + Alertmanager → MQTT (to be tested)
 - GUI → Tkinter / QT / Grafana
 - Multi-threading → Python threading / QT
- Distributed System Manager → MQTT Explorer
- DIAdem → Numerical Python
 - DataFinder (Indexing of Metadata) → Custom replacement based on e.g. SQLite
- **AF/CS++** → Pykka/PyAcdaq
- **CS/CS++:** Don't change, do not upgrade, freeze and migrate!

- Pykka: <https://pykka.readthedocs.io/en/stable/>
 - *Pykka is a Python implementation of the actor model. The actor model introduces some simple rules to control the sharing of state and cooperation between execution units, which makes it easier to build concurrent applications.*
 - **Inpiration**
 - *Much of the naming of concepts and methods in Pykka is taken from the Akka project which implements actors on the JVM. Though, Pykka does not aim to be a Python port of Akka, and supports far fewer features.*
 - *Notably, Pykka **does not** support the following features:*
 - *Supervision: Linking actors, supervisors, or supervisor groups.*
 - *Remoting: Communicating with actors running on other hosts.*
 - *Routers: Pykka does not come with a set of predefined message routers, though you may make your own actors for routing messages.*
- *Very similar to NI Actor Framework*

- Git: <https://git.gsi.de/EE-LV/ACDAQ/PyAcdaq>
- MQTT Explorer
- Example GUI
- UML class Diagrams



AcdaqSerDevSim: Code Examples



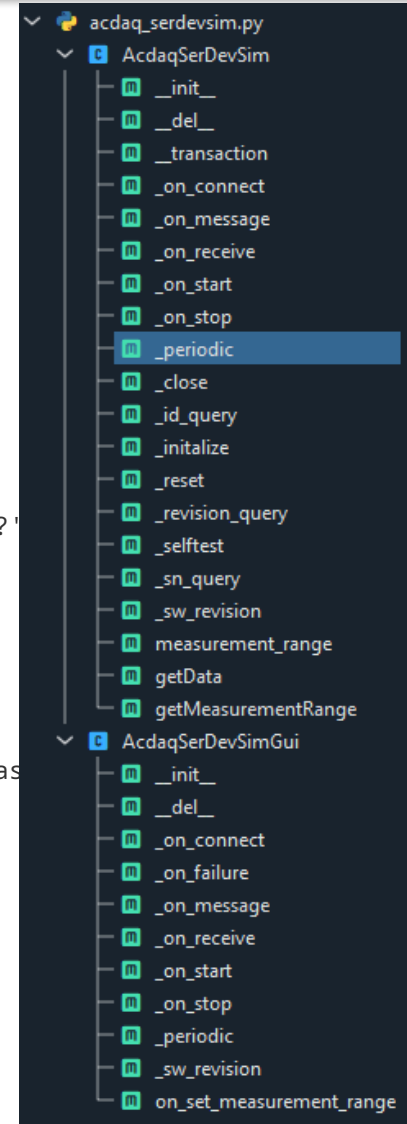
```
# Protected override method
def _initialize(self,
                  resource,
                  id_query_flag,
                  reset_flag,
                  reset_options,
                  selftest_flag):
    """
    Open serial connection and initialize instrument.

    Returns
    -----
    int
        Error code.
    """
    logging.debug(self._name +
                  ' SerDevSim _initialize() Initializing
connection.')
```

```
# Protected override method
def _periodic(self):
    """
    Implementation of periodic actions.

    Read data from instrument.

    Returns
    -----
    None.
    """
    try:
        self._data = int(self.__transaction('READ?'))
        logging.debug(self._name + " Data=,"
                      + str(self._data))
        if self._mqtt is not None:
            self._mqtt.publish_topic(self._name +
                                     '/Data', self._data,
                                     qos=0, retain=False, expand=False)
        except acdaq_exceptions.CommunicationTimeout as e:
            logging.error(self._name + ' SerDevSim
_periodic()
            Communication timed out for\ command:
            {}'.format(e.command))
            raise e
        pass
```



Tkinter & matplotlib

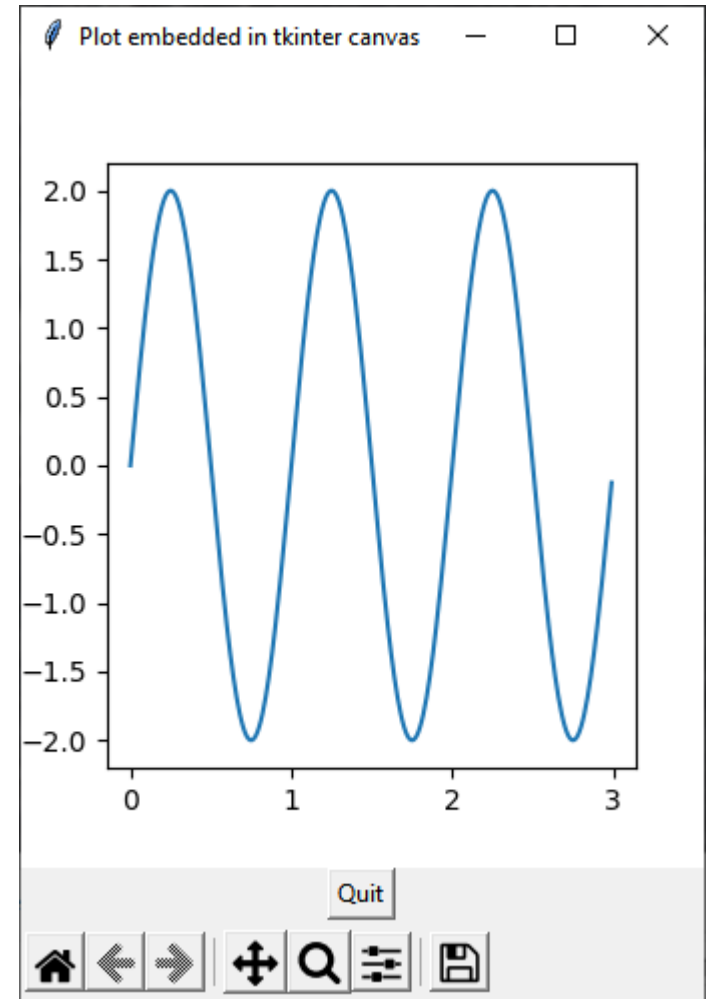
<https://youtu.be/2MQEHYjfrf8>

```
import tkinter as tk
from matplotlib.backends.backend_tkagg import
(FigureCanvasTkAgg, NavigationToolbar2Tk)
from matplotlib.backend_bases import key_press_handler
from matplotlib.figure import Figure
import numpy as np

def on_key(event):
    print('You pressed {}'.format(event.key))
    key_press_handler(event, canvas, toolbar)

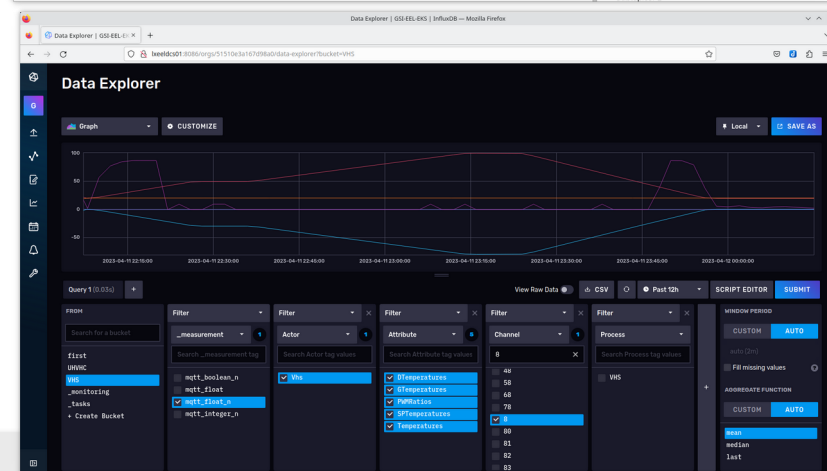
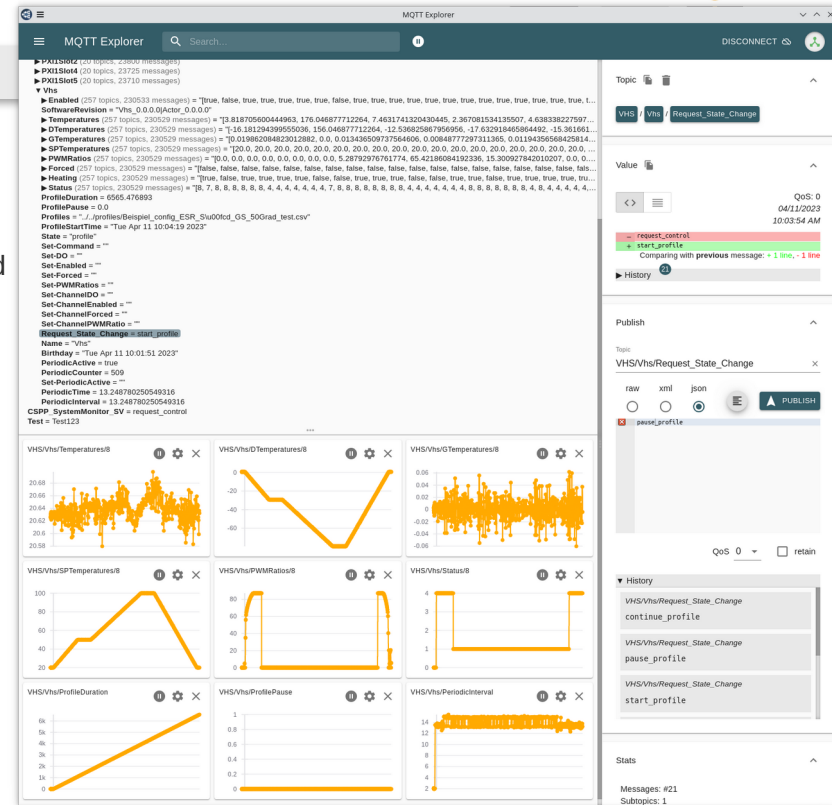
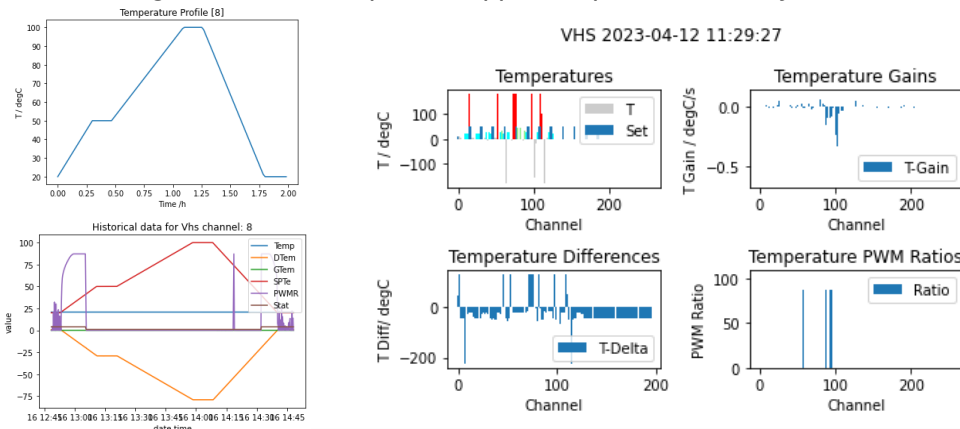
def quit():
    root.quit()
    root.destroy()
    pass

t = np.arange(0, 3, 0.01)
root = tk.Tk()
root.wm_title('Plot embedded in tkinter canvas')
fig = Figure(figsize=(5,4),dpi=100)
fig.add_subplot(111).plot(t, 2. * np.sin(2. * np.pi * t))
canvas = FigureCanvasTkAgg(fig, master=root) # A Tk drawing area
canvas.draw()
canvas.get_tk_widget().pack(side=tk.TOP, fill=tk.BOTH, expand=1)
toolbar = NavigationToolbar2Tk(canvas, root)
toolbar.update()
canvas.get_tk_widget().pack(side=tk.TOP, fill=tk.BOTH, expand=1)
canvas.mpl_connect('key_press_event', on_key)
button = tk.Button(master=root, text='Quit', command=quit)
button.pack(side=tk.BOTTOM)
tk.mainloop()
```



Vacuum Heating System (Bakeout)

- PyAcdaq/pykka based project for the GSI Vacuum Heating System (VHS) to be used at ESR and SIS.
- It is utilizing PXIe HW from NI. It measures temperatures from analog inputs and applies a fuzzy system to calculate the digital output pulse width ratio for the heating power supplies.
- It demonstrate a migration path from LabVIEW RT and DSC to
 - Python / pykka / PyAcdaq / *nidaqmx*
 - python-statemachine / simplful / paho-mqtt
 - numpy / pandas / matplotlib
 - Mosquitto / MQTT-Explorer / Telegraf / InfluxDB.
- Temperature Profile execution with no HW connected is working.
- Comissioning with connected power supplies is planned for May 2023.



- DIM: <https://dim.web.cern.ch/dim/>
- CS Framework
 - <https://wiki.gsi.de/CSframework/WebHome>
- Successful Feasability Study: Mobile Agent based on LVOOP (LabVIEW 8.5)
 - <https://wiki.gsi.de/foswiki/bin/view/NIUser/LVMobileAgentSystem>
 - <https://wiki.gsi.de/foswiki/pub/NIUser/LVMobileAgentSystem/DiplomarbeitFrederikBerck.pdf>
 - <https://forums.ni.com/t5/LabVIEW-Development-Best/Developing-a-Mobile-Agent-System-Using-LabVIEW-Object-Oriented/ta-p/3531213>
 - <https://forums.ni.com/t5/LabVIEW-Development-Best/Studying-the-Feasibility-of-an-Agent-System-Based-on-LabVIEW/ta-p/3531225>
- Actor Framework
 - <https://forums.ni.com/t5/Actor-Framework-Documents/tkb-p/7301>
- **CS++**
 - Scientific Report 2014 GSI Report 2015-1, 459 p. (2015)
<http://repository.gsi.de/record/184173/files/FG-GENERAL-41.pdf>

■ CS++

- H.Brand, 2015, *“Proposal for Zero Coupling Linked Network Actor Messages”*
<https://forums.ni.com/t5/Actor-Framework-Documents/CS-LNAMessage-pdf/ta-p/3498278>
- H.Brand, D. Neidherr „CS++ - NI Actor Framework-based Class Library“, *“Virtuelle Instrumente in der Praxis 2016 - Begleitband zum 21. VIP-Kongress”*, Rahman Jamal, Ronald Heinze (Hrsg.), VDE Verlag, ISBN 978-3-8007-4208-0
<http://sine.ni.com/cs/app/doc/p/id/cs-17505>
- H.Brand, D. Neidherr,... „Anwendung von AF/CS++ auf CompactRIO am Beispiel der GEM-Disc-Produktionsanlage für PANDA@FAIR“, *“Virtuelle Instrumente in der Praxis 2017 - Begleitband zum 22. VIP-Kongress”*, Rahman Jamal, Ronald Heinze (Hrsg.), VDE Verlag, ISBN 978-3-8007-4441-1
- H.Brand, D. Neidherr, 2018, *“Proposal for Asynchronous Callback Messages in the Actor Framework”*
<https://forums.ni.com/t5/Actor-Framework-Documents/Proposal-for-Asynchronous-Callback-Messages-in-the-Actor/ta-p/3837769>
- A.N. State, ..., H. Brand, ..., *“The slow control system of the FRS Ion Catcher”*
Nuclear Inst. and Methods in Physics Research, A 1034 (2022) 166772,
<https://authors.elsevier.com/c/1f2TacPqbl2QS>
- M. Jankowski et.al. *“Recent developments in β -NMR”*, ISOLDE Workshop and Users Meeting 2021
Indico: <https://indico.cern.ch/event/1073670/contributions/4630932/>
Book of Abstracts: <https://indico.cern.ch/event/1073670/book-of-abstracts.pdf>

- Brief vom 23. April 2021
 - „Empfehlungen zur Nutzung der Lizenzen“, H. Brand, EEL
- Um mindestens das finanzielle Risiko zu begrenzen, werden in Rücksprache mit unserer stellvertretenden Forschungsdirektorin Y. Leifels folgende Empfehlungen zur Nutzung der verfügbaren Lizenzen vorgeschlagen:
 - Entwickler sollen in ihren Aktivitäten nicht eingeschränkt werden.
 - Alle verfügbaren Module mögen eingesetzt werden, um Anwendungen möglichst effizient zu entwickeln.
 - Entwickler mögen auf modulare Software-Architekturen achten und mögliche Migrationsstrategien berücksichtigen, um einen Vendor-Lock zu vermeiden.
 - Anwendungen sollen möglichst in ausführbare Programme (Executable) kompiliert werden. Sie sollen nicht dauerhaft in der Entwicklungsumgebung laufen.
 - Für viele Programme ist eine kostenfreie Runtime-Lizenz ausreichend. Das kann helfen, die Anzahl der benötigten Lizenzen niedrig zu halten.
 - Manche NI Software Module benötigen für das Ausführen eines Executable eine Deployment Lizenz. Während der Vertragslaufzeit ist das unproblematisch, sie wird vom NI-VLM gewährt. Dabei wird je eine Entwicklungslizenz für den Computer gezählt. Nur bei Beendigung des Vertrags würden dann entsprechende einmalige Kosten anfallen, die aber deutlich niedriger sind als eine Entwicklungsumgebung und nicht verfallen.
 - Zusammen mit einem Versionskontrollsystem, z.B. Git (git.gsi.de) kann es auch helfen, eine geordnetes Release-Management zu organisieren.