

# Standardization of software device drivers implementation for data acquisition and timing devices in ITER CODAC Core System: Nominal Device Support

**mini!!!**

M. Ruiz<sup>1</sup>, J. Moreno<sup>2</sup>, A. de Gracia<sup>1</sup>, S. Melis<sup>2</sup>, S. Esquembri<sup>1</sup>, D. Sanz<sup>2</sup>, M. Astrain<sup>1</sup>, R. Lange<sup>3</sup>

<sup>1</sup> Instrumentation and Applied Acoustic Research Group, Technical University of Madrid, Madrid, Spain

<sup>2</sup> GMV Aerospace and Defence S.A.U., Isaac Newton 11, P.T.M. Tres Cantos, 28760, Madrid, Spain

<sup>3</sup> ITER International Organization St. Paul lez Durance, France

[mariano.ruiz@upm.es](mailto:mariano.ruiz@upm.es)

- Multiple platforms, chassis, devices....
- Multiple Control Systems (EPICS, TANGO, MARTe, RTF,...)
- Multiple device drivers
- Interface hell

The solution?

## Nominal Device Support 3 (NDSv3)



PXIe



X-Series



PXI6683



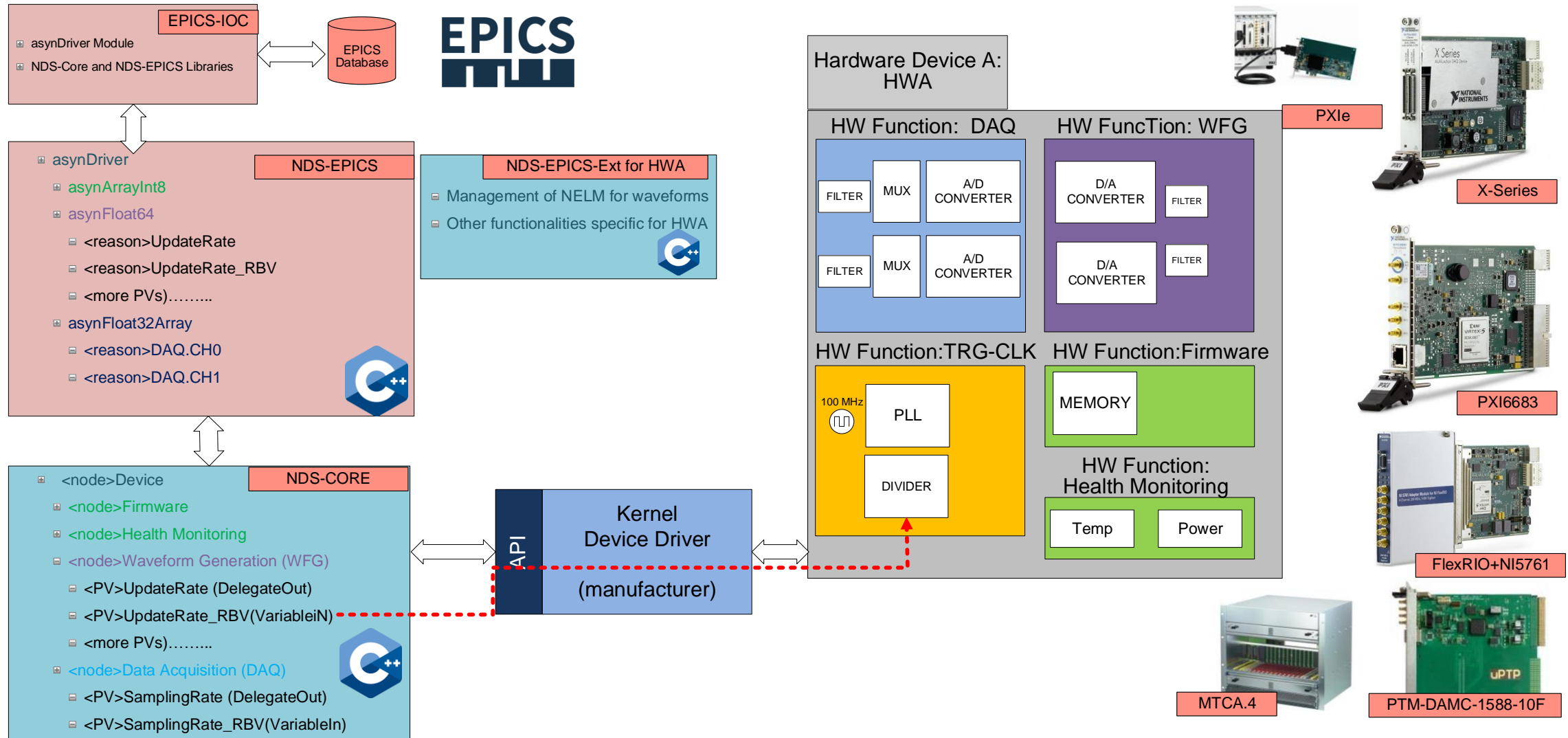
FlexRIO+NI5761



MTCA.4



PTM-DAMC-1588-10F



# Conclusions

- Integrated in ITER CODAC Core System 6.0 (NDSv 3.2.0.0)
- Drivers available for PXIe Devices: PXI6683 (timing card), PXIe6363 (X-Series DAQ Devices) and FlexRIO with analog input module NI5761
- Drivers available for MTCA devices: PTM1588 (timing card)
- Version with new functionalities in Q4 2019 (v3.3.0.0)
- Support for ITER Data Archiving Network and Synchronous Data Network

# Standardization of software device drivers implementation for data acquisition and timing devices in ITER CODAC Core System: Nominal Device Support

M. Ruiz<sup>1</sup>, J. Moreno<sup>2</sup>, A. de Gracia<sup>1</sup>, S. Melis<sup>2</sup>, S. Esquembri<sup>1</sup>, D. Sanz<sup>2</sup>, M. Astrain<sup>1</sup>, R. Lange<sup>3</sup>

<sup>1</sup> Instrumentation and Applied Acoustic Research Group, Technical University of Madrid, Madrid, Spain

<sup>2</sup> GMV Aerospace and Defence S.A.U., Isaac Newton 11, P.T.M. Tres Cantos, 28760, Madrid, Spain

<sup>3</sup> ITER International Organization St. Paul lez Durance, France

[mariano.ruiz@upm.es](mailto:mariano.ruiz@upm.es)

## Thanks for your attention!!

This work was supported in part by the Spanish Ministry of Economy and Competitiveness, Projects Nº ENE2015-64914-C3-3-R and Madrid regional government (YEI fund), Grant Nº PEJD-2018-PRE/TIC-8571.

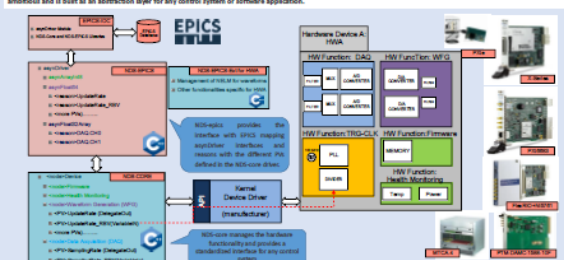


Standardization of software device drivers implementation for data acquisition and timing devices in ITER CODAC Core System: Nominal Device Support

M. Ruiz<sup>1</sup>, J. Moreno<sup>2</sup>, A. de Gracia<sup>1</sup>, S. Melis<sup>2</sup>, S. Esquembri<sup>1</sup>, D. Sanz<sup>2</sup>, M. Astrain<sup>1</sup>, R. Lange<sup>3</sup>

<sup>1</sup>Instrumentation and Applied Acoustic Research Group, Technical University of Madrid, Madrid, Spain  
<sup>2</sup>GMV Aerospace and Defence S.A.U., Isaac Newton 11, P.T.M. Tres Cantos, 28760, Madrid, Spain  
<sup>3</sup>ITER International Organization St. Paul lez Durance, France  
[mariano.ruiz@upm.es](mailto:mariano.ruiz@upm.es)

**SUMMARY**  
 The standardization of device drivers implementation in distributed control systems is essential to reduce the development and integration effort. Nowadays, there are multiple distributed control systems and high-level applications used in the scientific community. Due to the lack of standardization in the interfaces with the device drivers, it is required to integrate each hardware device separately in each control system. However, a clear model of how different software applications should manage the device drivers could reduce development efforts and ease the maintainability of such complex systems. The Nominal Device Support Initiative is an open source project, for the standardization of data acquisition and timing devices. The first implementations of NDS (version 1.0 and 2.0) were oriented to simplify the implementation of EPICS device supports. NDS v3 is more ambitious and is built as an abstraction layer for any control system or software application.



**MAIN FEATURES OF NDS-CORE LAYER**


- Software layer implemented in C++ in charge of interfacing with the hardware using the device driver provided by the manufacturer.
- The driver manages the functionality using a hierarchical organization in nodes, called NDS-nodes. Each NDS-node manages a specific hardware block (i.e. firmware, ADC, DAC, Timing, clocking, triggering, timestamping, frame timing event, etc.).
- An NDS-node provides a list of defined process variables in charge of the configuration and management of the specific hardware functionality.
- The NDS-Pub are identified as Variable/Out and Diagnostic/Out. NDS-Pub implement functions to read and write to specific values. Additionally, Diagnostic/Pub trigger the execution of specific methods when are written or read.
- NDS-core provides method to subscribe and replicate Pub. This option allow to share data among different NDS-Device Drivers.
- An NDS-Device driver is implemented using NDS-core and is available for the user as a library (shared or static) to be used in any application.
- NDS-core layer has been tested using google test. A simplified test control system is provided with the test code.

**MAIN FEATURES OF NDS-EPICS LAYER**

- A unique software layer implemented in C++ to interface multiple NDS-core device drivers to EPICS control system. This approach simplifies the code maintainability and testability.
- EPICS Pub are mapped (connected) to NDS-core Pub. Every action to read and write to the asyDriver interface triggers the specific read/write function in the NDS-core Pub.
- NDS-core Push functions are mapped to asyDriver interrupts to send data to EPICS.
- The user only needs to configure the EPICS database file using templates and substitutions file to manage a specific NDS-Device driver.
- The NDS-EPICS layer can be extended by the user to support additional functionalities. Profiles are read by the NDS-Device driver (i.e. use of number of elements (NEM) for waveforms).
- NDS-EPICS tests has been implemented using PyEPICS.

**ACKNOWLEDGEMENTS**

• Integrated in ITER CODAC Core System 6.0 (NDS-3.0.0.0)  
 • Drivers available for P30a-Devicer: P30S03 (timing card), P30aD03 (i-Center DAQ Device) and P30aI03 with analog input module N3T01  
 • Drivers available for MTC-A Device: PFM1503 (timing card)  
 • Version with new functionalities in Oct 2019 (v3.0.0.0)  
 • Support for ITER Data Archiving Network and Synchronous Data Network



Poster location: P/4-5  
16/05/2019