

Control system based on a
Highly
Abstracted and
Open
Structure



WP2

C. Bisegni

WP2 Task

what has been done during project

- Common Layer - New Development, Debug and Improvement
- CU Layer - Improvement
- Data Service - New Development
- Metadata Server - New Development
- WAN Proxy - New Development

Common Layer

what has been done during project

- DirectIO, the !CHAOS high performance data transfer channel,
 - ZMQ Direct IO Driver
- Synchronous HTTP RPC interface

CU Layer

what has been done during project

- Improvement on overall toolkit
- UnitServer now permit to encapsulate on the command for load a control unit, to attach a lis too command to forward to CU after it is loaded.

Data Service

is one of new nodes that has been developed during
project

- Use Direct IO for fast data transfer
- Driver abstraction for Database, Memory Cache, FileSystem Storage
- Answer to “**Get Last**” and “**Push New**” data query (from CU and other nodes)
- query on a single device as “*Proof of Concept*”

Data Service

what is to improve or add to be usable on CDS

- lack of ageing on data, so the data is never erased.
- query are asynchronous! This is not a good design for this kind of api, and bring a lot of problem. Needs to be reimplemented in synchronous way.

Metadata Service

All-in-One Java version has been replaced by Client-Server structure

- Full refectory of CHAOS constant.
- MetadataService & MetadataServiceClientLib
- MetadataService perform the api execution and a lot of operation are executed in batch environment(the same of the slow control unit)
- MetadataServiceClientLib:
 - has the proxy for all the server side api. This abstract the RPC data pack creation.
 - has also and advanced monitor system, that permit to attach handler to a variable of an NodeID(aka device ID) for track value changes, choosing the time slot(multiplier for base time offset 100ms, was can be updated)

Metadata Service

Metadata Server permit to 1-2

- Create a New Unit Server and Fill With Control unit type
- Register a already configured or new, Unit Server
- Configure a Control Unit Instance
 - NodeID(aka device_id)
 - Load Parameter
 - Driver
 - Input Dataset Startup configuration
 - auto-start, auto-init, auto-load

Metadata Service

Metadata Server permit to 2-2

- Open a Control Unit Editor that permit to control the target control unit for:
 - Load, Unload, Init, Deinit, Start and Stop
 - View and track the output dataset attribute value
 - View and change the input dataset attribute value
 - visualise the “real” push rate
 - change the desiderated push rate
 - Manage the command (for SLCU) and:
 - create, edit and remove template for a command
 - submit a template (valorising the mandatory input parameter)

Metadata Service

Chaos Control Studio

- Base on QT5.5 and MetadataServiceClientLib
- Permit, grafically, to manage al metadata previously described
- Permit to monitor output and input dataset attribute for a control unit except, string and buffer types.



!CHAOS

thanks you