

Évolution des outils de contrôles des applications et de collection de données de Virgo

Workshop R&Ds - Développements Instrumentaux / Virgo-ET
Marseille 5 mars 2024

emmanuel.pacaud@lapp.in2p3.fr – LAPP

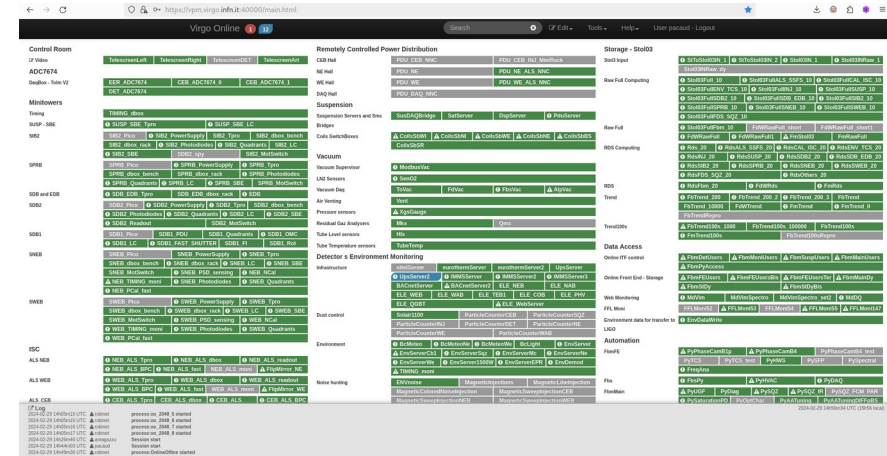
Bibliothèques de bases

- **Cm : communication inter process**
- **Fr : lecture/écriture du format Frame**
- **Cfg : configuration, messages Cm standards, boucle principale, états standards**
- **Fd : envoi/réception de flux de données au format Frame**

- **Communication point à point**
- **Messages structurés**
- **Langage C**
- **NameServer**
 - Mise en relation des process
 - Assure l'unicité du nom des process

Cm : les bons points

- Simple
- Fiable
- Efficace
 - 674 process surveillés par VirgoProcessMonitoring (Vpm)

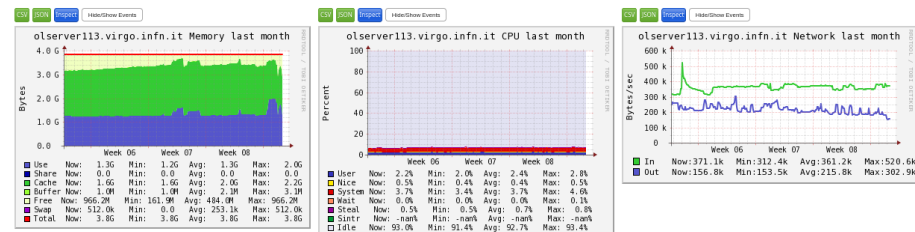


SNEB_dbox_rack User Interface User pacaud - Logout

Name	SIN	Firmware	Type	Status
SNEB_DBOX_SBE	127	v4r5	daqbox	Connected
SNEB_ADC_SBE0	45	v4r1	adc2378	Connected
SNEB_ADC_SBE1	49	v4r1	adc2378	Connected
SNEB_DAC_SBE0	102	v2r0	dac1955	Connected
NEB_DEMOD_ENV	v3r2 - v3r4		demodulation	Connected
SNEB_DBOX_LC	34	v4r4	daqbox	Connected
SNEB_ADC_MONIO	54	v4r1	adc2378	Connected
SNEB_DAC_LC_LVDT_out	31	v2r0	dac1955	Connected

Mezzanine Camera NEB_MezzCAM

Camera	Shutter	Power
NE_Cam	Unknown	Open Close On Off
ALS_NE_Cam1	Unknown	Open Close On Off
ALS_NE_Cam2	Unknown	Open Close On Off



Cm : les points faibles

- **Pas thread safe**
- **Gestion de connexions multiples compliquée**
- **Commande/acquitement compliqué**
- **Structure des messages figée**
 - Contournement possible dans certains cas

Critères pour la sélection d'un remplaçant

- **120 applications doivent être portées**
- **Utilisable depuis C, C++, python**
- **Possibilité de support de la nouvelle solution et des applications Cm/Cfg dans Vpm pour faciliter la transition**
- **Multithread**
- **Open source**
- **Contrôle/commande et collection de données**
- **Protocole sécurisé**

Tango

- **Utilisé initialement pour le contrôle du vide et des suspensions**
- **En addition de Cm/Cfg, pas d'effort pour porter le reste des logiciels Virgo**
- **Difficulté de gestion des différents services, plus de compétences dans la collaboration**
- **Abandon pour le contrôle des suspensions**

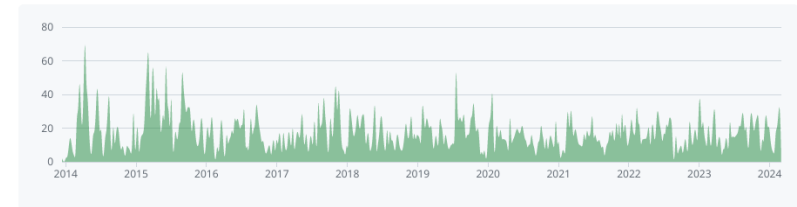
OPC Unified Architecture (OPC UA)

- **OPC foundation**
- **IEC Standard 62541**
- **Support des modes client-server et publish-subscribe**
- **Transmissions sécurisées (chiffrement, sommes de contrôle)**
- **Multi-plateformes**
- **Multiples implémentations**
- **Sérialisation binaire et JSON**

Name	Language	License	Client/Server	Link
open62541	C	MPL-2.0	Client and Server	http://open62541.org/
UA.NET Standard	C#	GPL, RPC for OPC Foundation Members	Client and Server	https://github.com/OPCFoundation/UA-.NETStandard
node-opcua	JavaScript	MIT	Client and Server	http://node-opcua.github.io/
FreeOpcUa	C++	GPL	Client and Server	http://freeopcua.github.io/
BaOpcUa	Lua	GPL + commercial license. Lua source code can be found inside mako_opc	Client/Server	https://realtimelocic.com/products/opc-ua/
Python FreeOpcUa	Python	GPL	Client and Server	https://github.com/FreeOpcUa/opcua-asyncio
OpenOpcUa	C++	Cecil-C, source code access costs a one-time fee for non commercial use and a paid support and maintenance for commercial use.	Client and Server	http://www.openopcua.org/
OpenScada UA Interface	C++	GPL	Server	http://oscada.org/websvn/...
ASNeG	C++	Apache	Server and Client	Git Repository
Eclipse Milo	Java	Eclipse Public License	Stack / Client / Server	https://github.com/eclipse/milo
opcua4j	Java	Creative Commons 3.0 BY-SA, depends on redistributable jar-files from the OPC Foundation	Server	https://code.google.com/p/opcua4j/
uafl	C++/ Python	GNU Lesser General Public License	Client (wrapper over proprietary sdk)	https://github.com/uafl/uafl
opc-ua-client	C#	MIT	Client	https://github.com/convertersystems/opc-ua-client
UAACL/CS	C#	GPL	Client/Server (wrapper over proprietary sdk)	https://gitlab.com/falko.wiese/uacl_cs
UAACL/CP	C++	GPL	Server(wrapper over proprietary sdk)	https://gitlab.com/falko.wiese/uacl_cpp
S2OPC	C	Apache	Client/Server	https://gitlab.com/systemrel/S2OPC
OPC UA Player	Java	MIT	Server Simulation	https://github.com/MileBaumajjer/OPCUA-Player
opcua	Rust	MPL-2.0	Client and Server	https://github.com/locka99/opcua
opcua	Golang	MIT	Client	https://github.com/gopcua/opcua
LibUA	C#	Apache 2.0	Client + Server	https://github.com/naufal/LibUA
opcua	TypeScript	MIT	Client	https://github.com/HBM/opcua
Apache PLC4X	Java	Apache	Client	https://plc4x.apache.org/users/protocols/opc-ua.html (prior PLC4X release 0.9 it was wrapper around Eclipse Milo. Starting from 0.9 it is standalone client implementation)

Une implémentation d'OPC-ua : Open62541

- **Licence libre (MPL v2.0)**
- **Implémentée en C**
- **Léger (profile minimal utilisable sur un micro-contrôleur type ESP32)**
<https://github.com/Pro/open62541-esp32>
- **API client et serveur**
- **Développement soutenu depuis 10 ans**
- **Une dizaine de contributeurs en 2023**



Application générique qui permet d'explorer ce qui est publié par un serveur opcua

The screenshot displays the Unified Automation UaExpert interface. The main window is titled "Unified Automation UaExpert - The OPC Unified Architecture Client - NewProject*". The interface is divided into several panes:

- Project:** Shows a tree view with "Project", "Servers" (containing "Local test@localhost"), and "Documents" (containing "Data Access View").
- Address Space:** Shows a tree view with "No Highlight", "Root", "Objects" (containing "Frame"), "Server", "Types", and "Views".
- Data Access View:** A table showing data from the server. The table has columns: #, Server, Node Id, Display Name, Value, Datatype, Source Timestamp, and Server T. The first row shows: 1, Local ..., NS1[String]..., Frame, 6b00000000..., ByteString, 15:23:26.645, 15:23:26.645.
- Attributes:** A table showing attributes for the selected node. The table has columns: Attribute and Value. The attributes listed are: Namespaceindex (1), IdentifierType (String), Identifier (frame), NodeClass (Variable), BrowseName (1, "frame"), and DisplayName ("en-US").
- References:** A table showing references for the selected node. The table has columns: Reference and Target DisplayName. The reference listed is: HasTypeDefin... BaseDataVariableType.
- Log:** A list of messages with columns: Timestamp, Source, Server, and Message. The messages include connection status, browse operations, and attribute reading.

À faire

- **S'assurer que les performances de transfert de données sont suffisantes**
- **Réimplémenter un NameServer**
- **Implémenter une couche d'émulation de l'API Cm/Cfg pour la transition**
- **Développer une API moderne pour standardiser l'implémentation des process Virgo**
 - États standard des applications
 - Métadonnées (liste des canaux produits par ex.)
 - Canaux de monitoring