

LabVIEW et OPCUA

Interfaçage avec un nouveau standard industriel



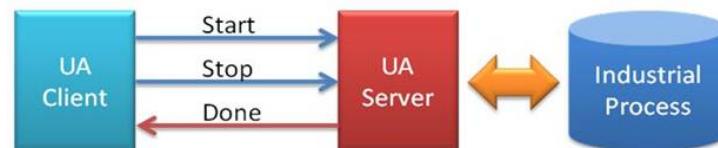
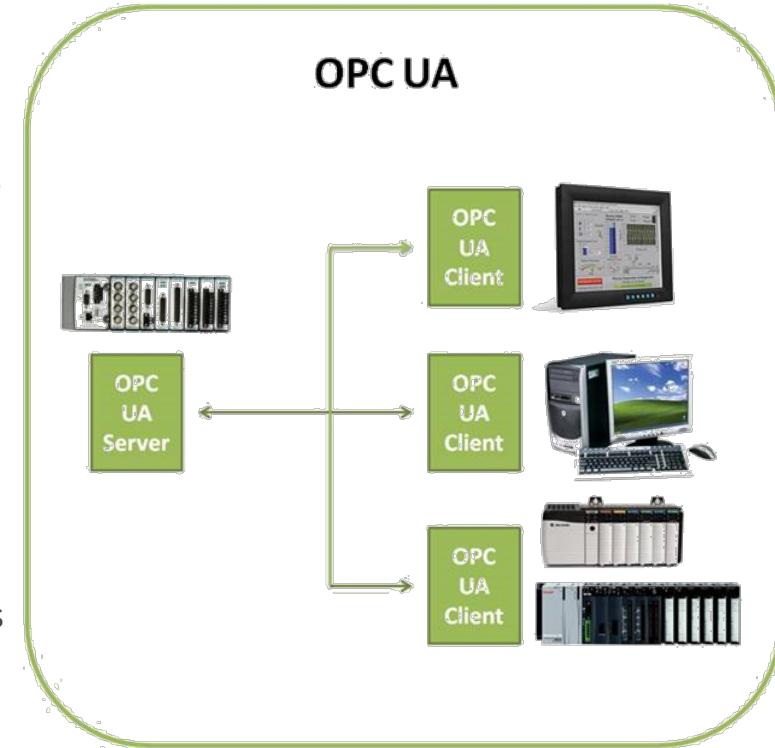
OPCUA késako?

(rien à voir avec OPC DA/HA...)

Généralités sur OPCUA



- ▶ Un standard défini par des fabricants et vendeurs de matériels/logiciels de control industriel
- ▶ Un framework cohérent, sécurisé et fiable orienté objet et multi-plateformes permettant d'accéder aux données et événements en temps réel et de façon historisée, à différents niveaux (aggrégation)
 - ▶ Des capteurs/actuateurs aux applications de control centralisées,
 - ▶ Des automates programmables industriels aux cartes électroniques embarquées.
- ▶ Indépendant des plateformes
 - ▶ Différents OS supportés pour les parties clientes et serveurs d'OPCUA => interopérabilité
 - ▶ Implémentation multi-languages (C, C++, JAVA)
- ▶ La plupart des matériels actuels respectent les "OPCUA device profiles" les plus basic du standard OPCUA => fonctionnement à la "PLC like"

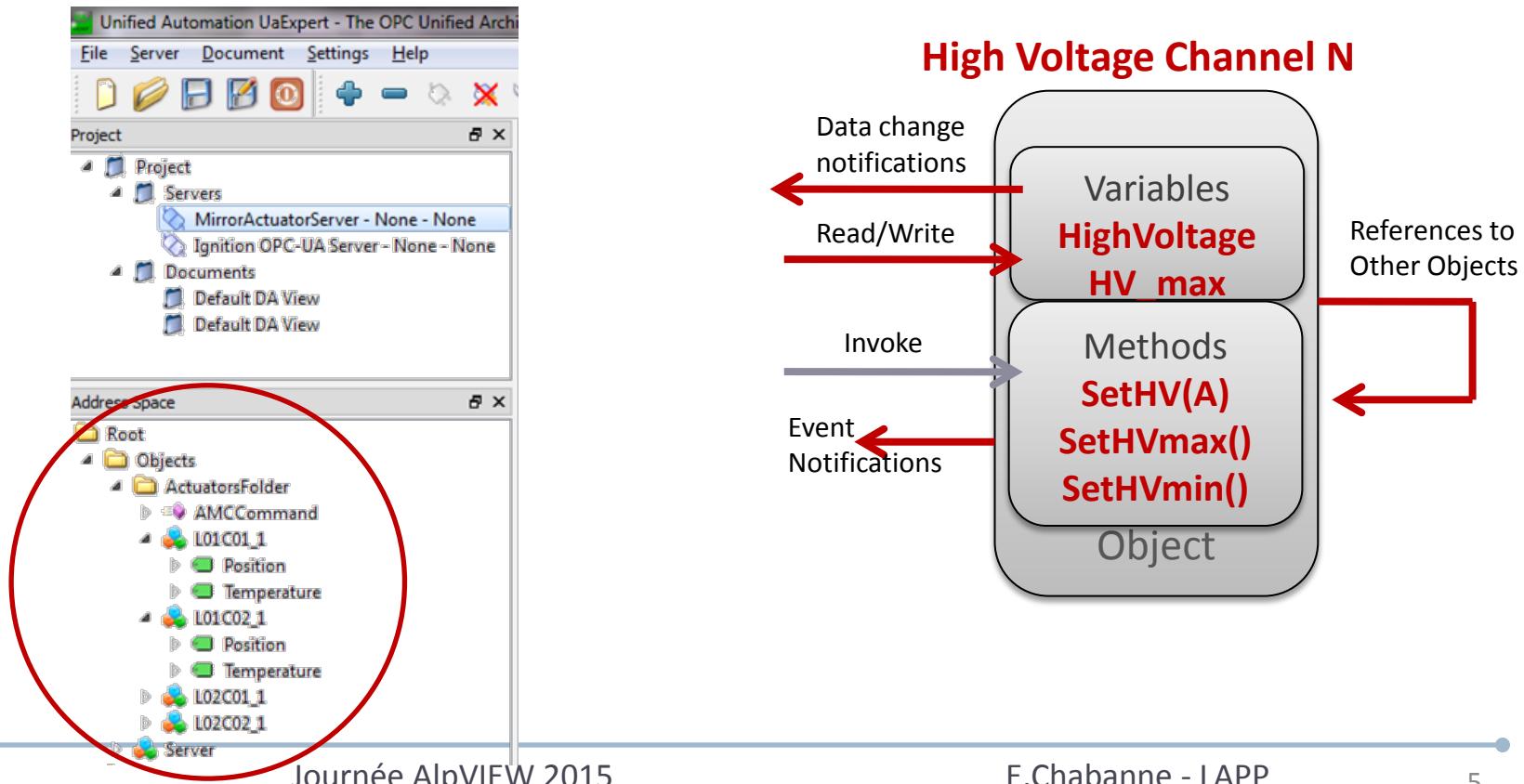


Généralités sur OPCUA

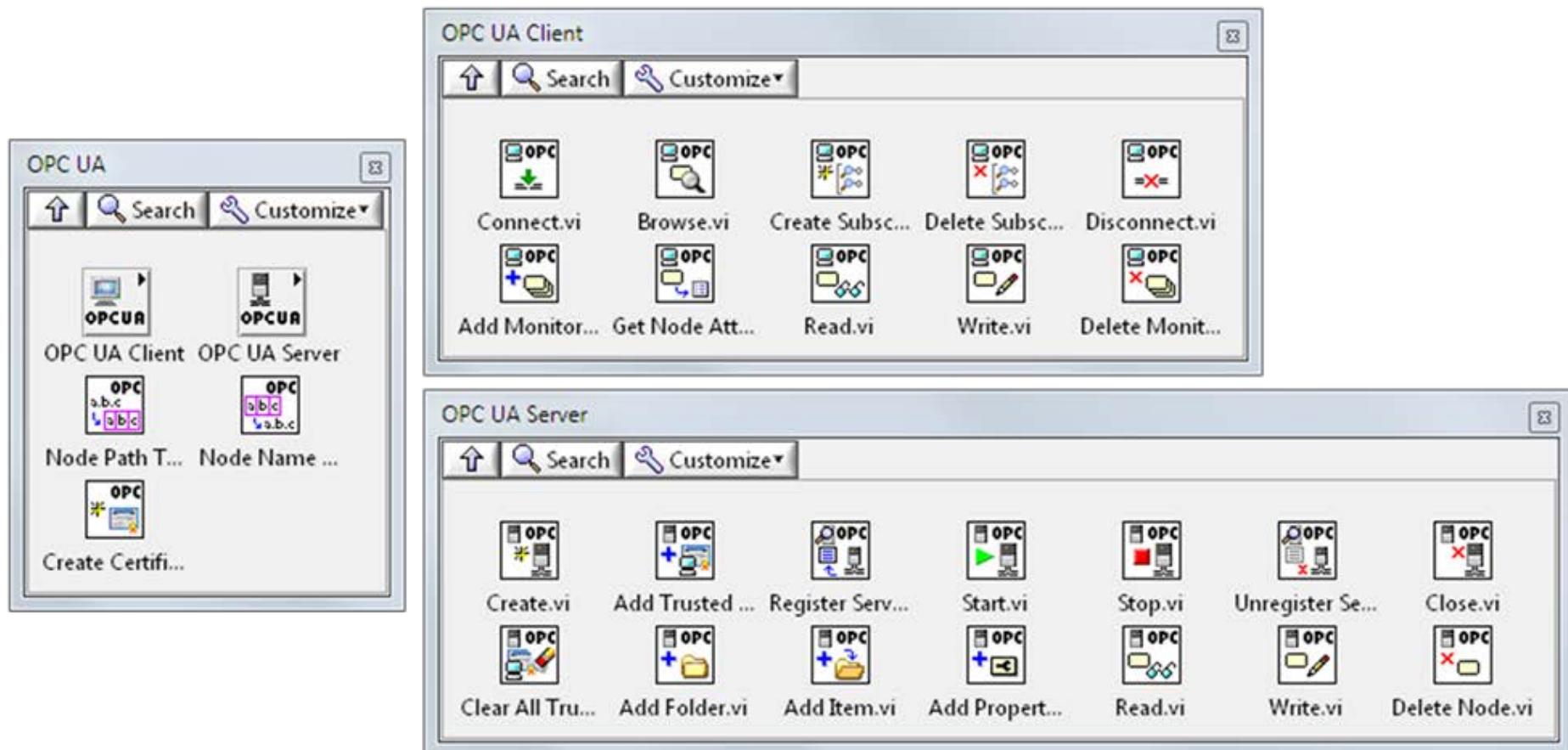
- ▶ **Testé sur des plateformes embarquées**
 - ▶ java et C++ sur des CPU ARM et X86
- ▶ **Canal de communication fiable et simplifié**
 - ▶ Les clients & serveurs font directement partie du système de command & control
 - ▶ Support natif de l'encryptage sécurisé (RSA 128 and 512bits)
 - ▶ Possibilité de redondance des serveurs
- ▶ **Architecture Orientée Service (SOA)**
 - ▶ Services génériques pour parcourir et interroger le NameSpace du serveur, les données en lecture/écriture, et publication/souscription aux événements et changements sur les données.
- ▶ **L'OPC foundation a récemment (mars 2015) ouvert les spécifications du standard.**
- ▶ **Quelques initiatives open sources en émergence**

Concept objet dans OPCUA

- ▶ **Framework Orienté Object**
 - ▶ Chaque nœud OPCUA est un objet qui peut exposer ses variables et méthodes aux autres nœuds
- ▶ **Tous les nœuds sont organisés à l'intérieur d'un 'Address Space' / 'NameSpace'**
 - ▶ Moyen standard de servir les objets entre serveurs et clients



OPCUA dans LabVIEW

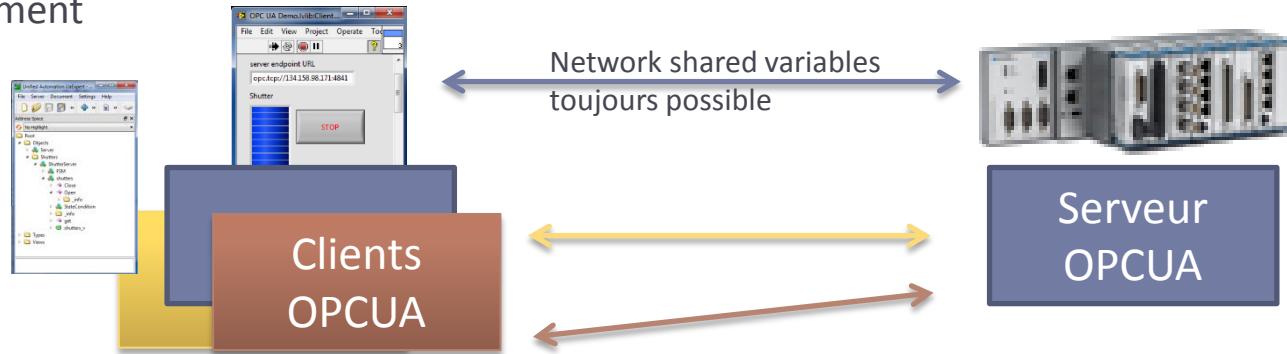


Disponible dans le module DSC pour les cibles Windows
dans le module Real-Time pour les cibles LabVIEW Real-Time.

2 possibilités d'architecture d'accès aux données

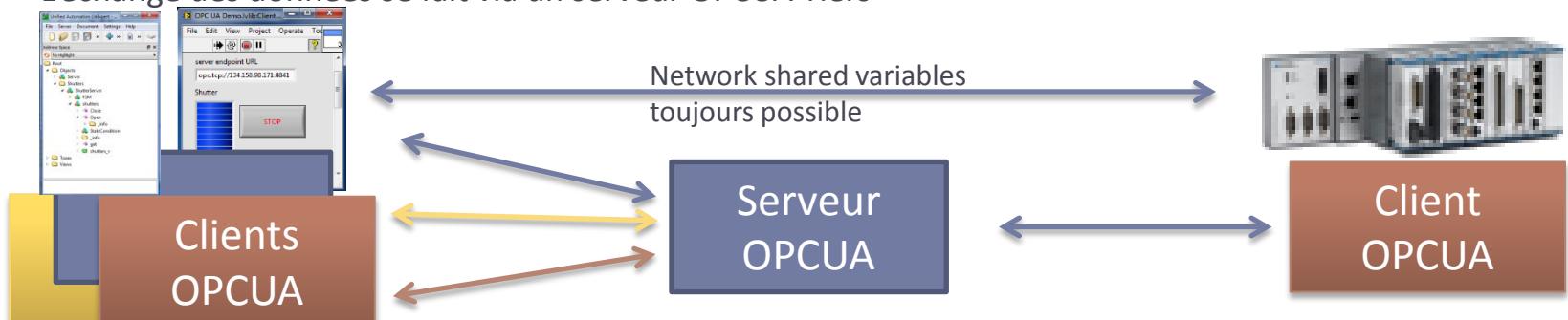
► Mode Client – Server

- ▶ Le producteur de données instancie un server OPCUA
- ▶ les consommateurs (clients OPCUA) viennent chercher les données ou s'abonner à leur changement



► Mode Client – Server – Client

- ▶ Le producteur et les consommateurs de données sont des clients OPCUA
- ▶ L'échange des données se fait via un serveur OPCUA Tiers



Exemples de clients OPCUA accédant aux même données

The screenshot shows the Unified Automation UaExpert interface and a LabVIEW OPCUA test client interface. The UaExpert window displays a 'Default DA View' table with data points, and a 'Log' window showing server activity. The LabVIEW window shows a plot of Amplitude vs. Time and a control panel with a 'STOP' button and status indicators.

Default DA View (UaExpert):

#	Server	Node Id	Display Name	Value
1	Unified Autom...	NS2[String]Device.Node.Set Point	Set Point	1000
2	Unified Autom...	NS2[String]Device.Node	Node	610
3	Unified Autom...	NS0 Numeric[2255	NamespaceArray	{http://opcfoundation.org/UA/,"
4	Unified Autom...	NS0 Numeric[2254	ServerArray	{'OpcUa Server'}
5	Unified Autom...	NS0 Numeric[2271	LocaleIdArray	{'en'}
6	Unified Autom...	NS0 Numeric[2258	CurrentTime	
7	Unified Autom...	NS0 Numeric[2259	State	0

Log (UaExpert):

Timestamp	Source	Server	Message
13/09/2013 16:5...	TypeCache	Unified Autom...	Translate returned error BadNodeIdUnknown for BrowsePath 'Icon' on NodeId NS0 Numeric 2009
13/09/2013 16:5...	TypeCache	Unified Autom...	Translate returned error BadNodeIdUnknown for BrowsePath 'Icon' on NodeId NS0 Numeric 2010
13/09/2013 16:5...	AddressSpaceM...	Unified Autom...	Browse succeeded.
13/09/2013 16:5...	TypeCache	Unified Autom...	Description =
13/09/2013 16:5...	TypeCache	Unified Autom...	DisplayName = Int32
13/09/2013 16:5...	TypeCache	Unified Autom...	BrowseName = 1:Int32
13/09/2013 16:5...	TypeCache	Unified Autom...	Read succeeded.
13/09/2013 16:5...	TypeCache	Unified Autom...	Reading type info of NodeId NS0 Numeric 6
13/09/2013 16:5...	Attribute Plugin	Unified Autom...	Read attributes of node 'NS2[String]Device.Node.Set Point' succeeded [ret = Good].
13/09/2013 16:5...	Reference Plugin	Unified Autom...	Browse succeeded.

OPC UA Demo.lvlib:Client.vi (LabVIEW):

server endpoint URL: `opc.tcp://localhost:4841`

Plot 0: Amplitude vs. Time

Device.Node.Set Point: 1000

error out: status: 1, code: -356612, source: NI OPC UA Client.lvlib:Connect.vi

STOP

Journée AlpVIEW 2015

E.Chabanne - LAPP

8

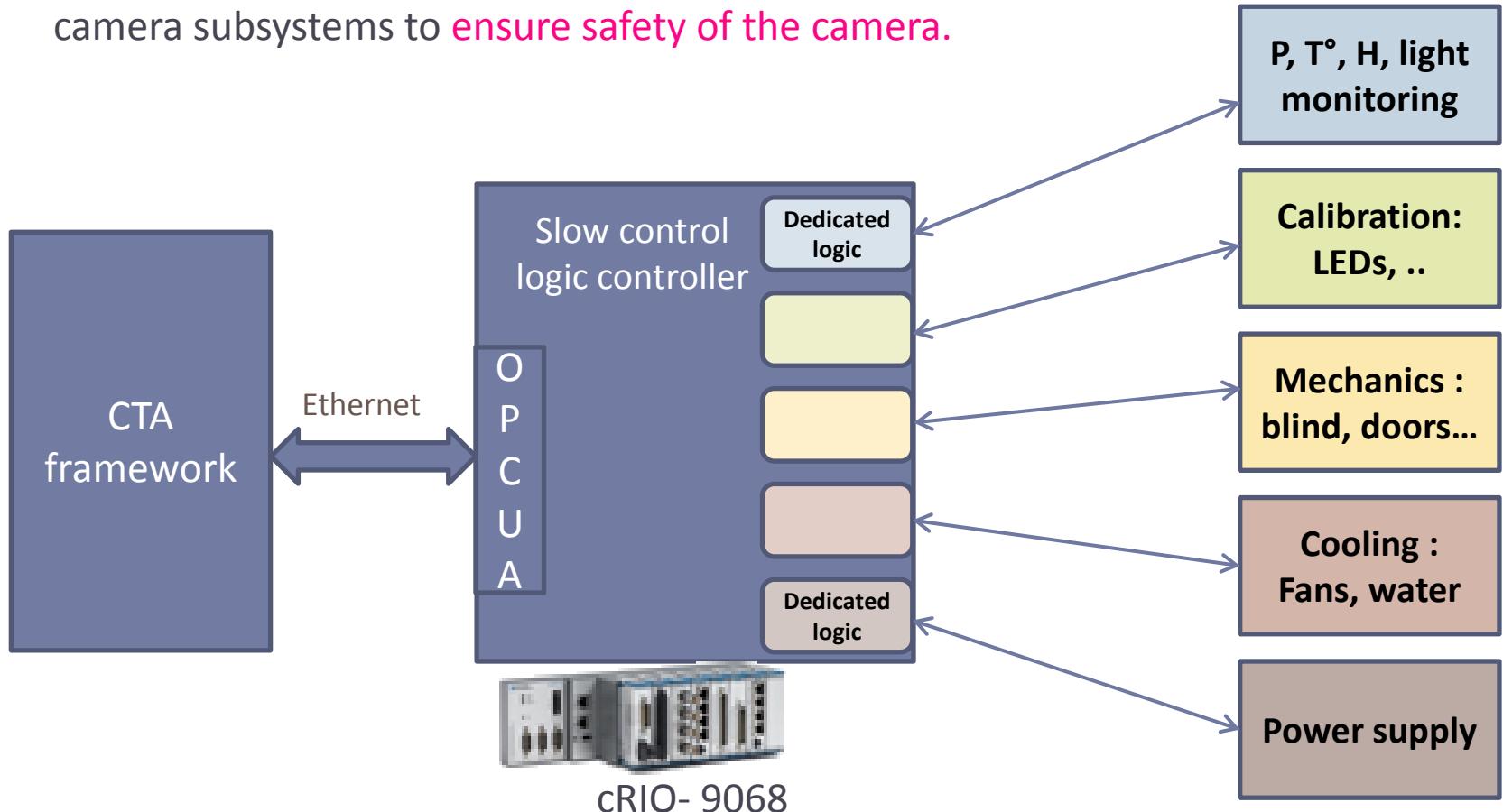
A large, white, segmented dish antenna, likely a Cherenkov Telescope Array (CTA) dish, is shown on a grassy hillside. The dish is mounted on a complex metal truss structure. In the background, other similar dishes and a road are visible under a clear sky.

Un exemple dans le projet CTA Cherenkov Telescope Array

Embedded Camera Controller

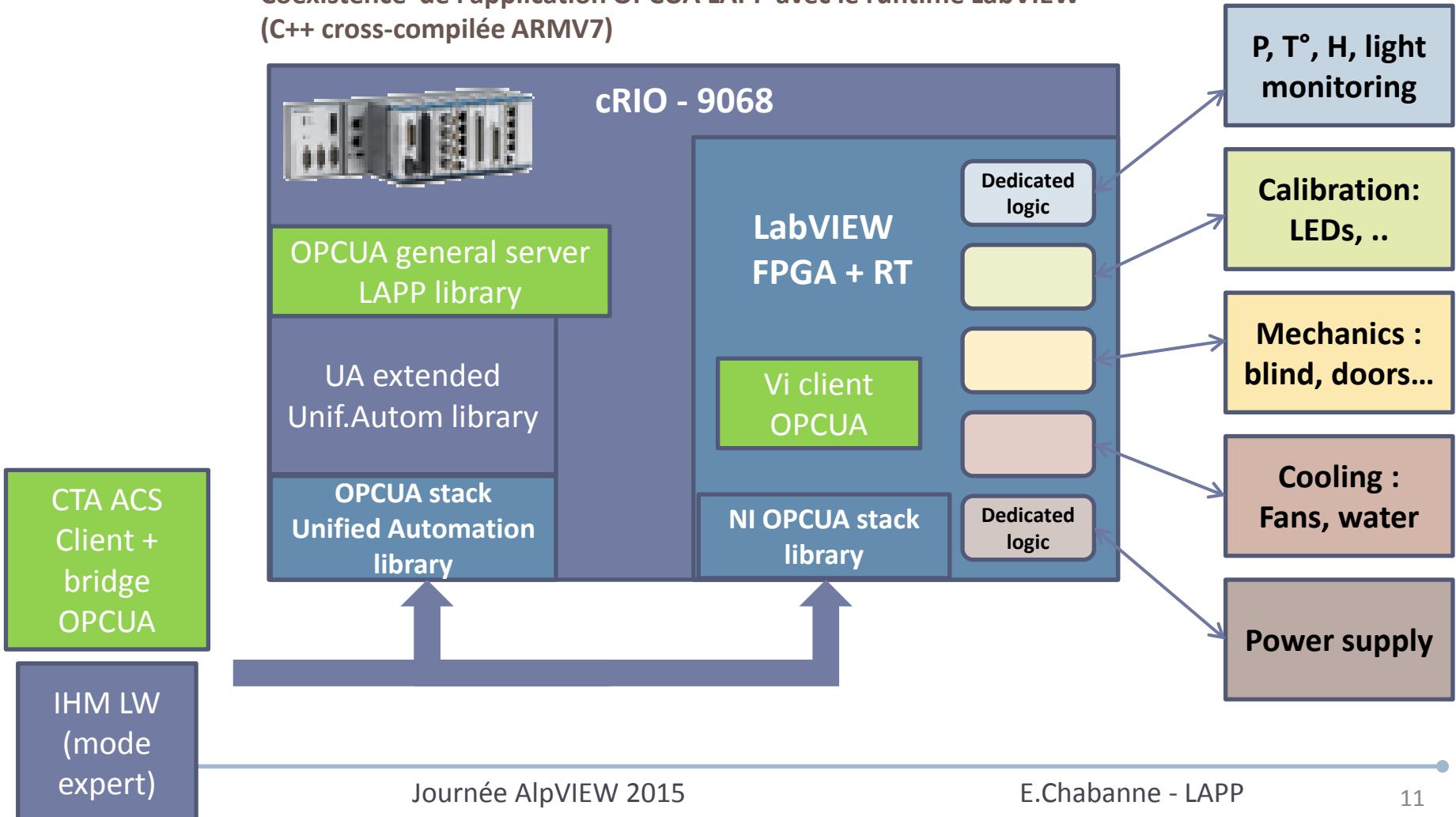
Bloc diagram du Slow control d'un type de caméra

SC logic controller monitors and controls different camera subsystems to **ensure safety of the camera**.



Organisation interne

Coexistence de l'application OPCUA LAPP avec le runtime LabVIEW
(C++ cross-compilée ARMV7)



134.158.98.94 - LAPP-ECC - SSH Secure Shell

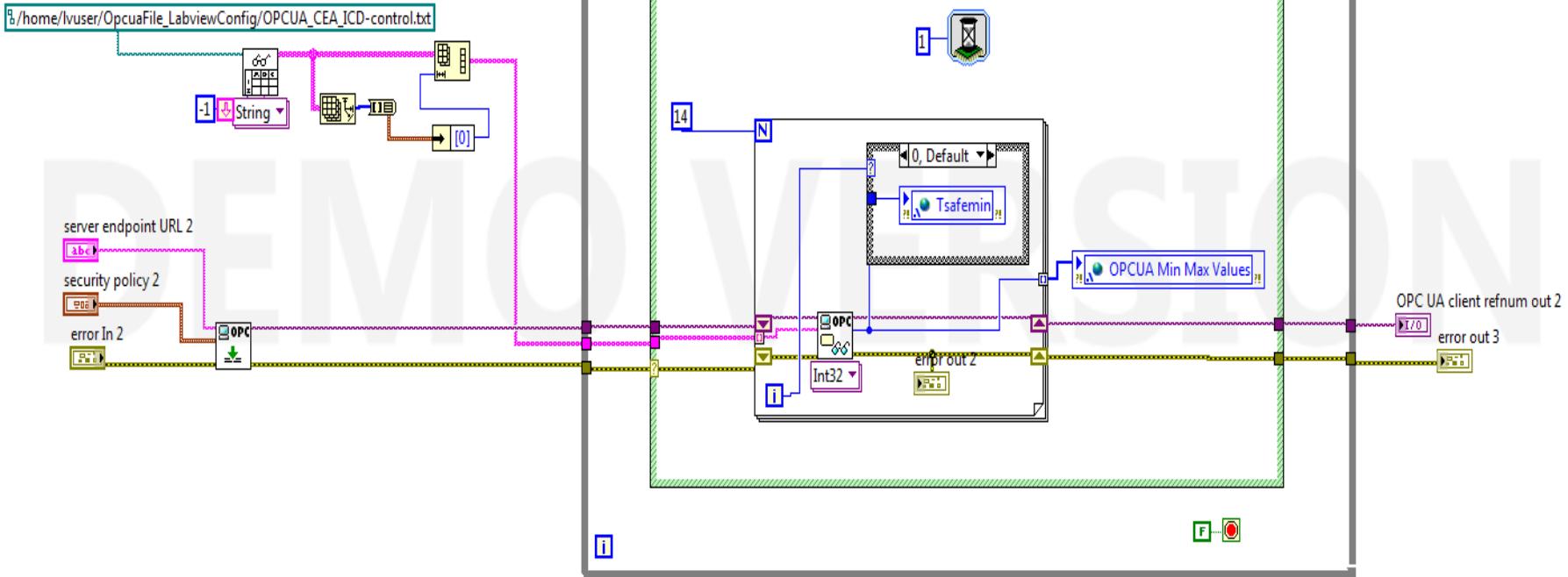
File Edit View Window Help

Quick Connect Profiles

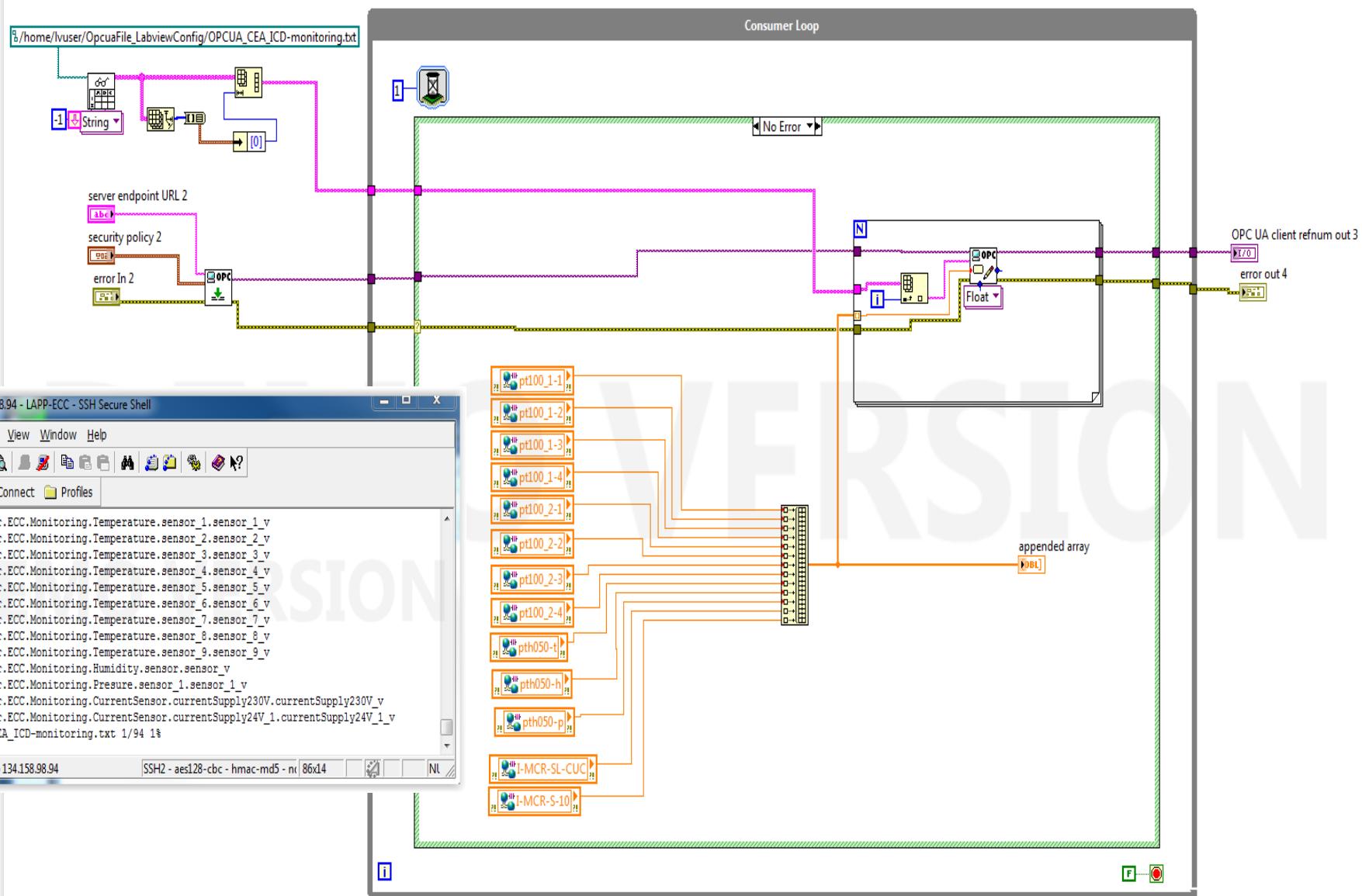
```
MOS_Server.ECC.Control.SafeMode.TSafemin_v
MOS_Server.ECC.Control.SafeMode.TSafemax_v
MOS_Server.ECC.Control.ReadyMode.TReadymin_v
MOS_Server.ECC.Control.ReadyMode.TReadymax_v
MOS_Server.ECC.Control.ReadyMode.Pmin_v
MOS_Server.ECC.Control.ReadyMode.Pmax_v
MOS_Server.ECC.Control.ReadyMode.Hmin_v
MOS_Server.ECC.Control.ReadyMode.Hmax_v
MOS_Server.ECC.Control.ReadyMode.Imin230V_v
MOS_Server.ECC.Control.ReadyMode.Imax230V_v
MOS_Server.ECC.Control.ReadyMode.Imin400V_v
MOS_Server.ECC.Control.ReadyMode.Imax400V_v
MOS_Server.ECC.Control.ReadyMode.Imin24V_v
- OPCUA_CEA_ICD-control.txt 1/20 5
```

Connected to 134.158.98.94 SSH2 - aes128-cbc - hmac-md5 -

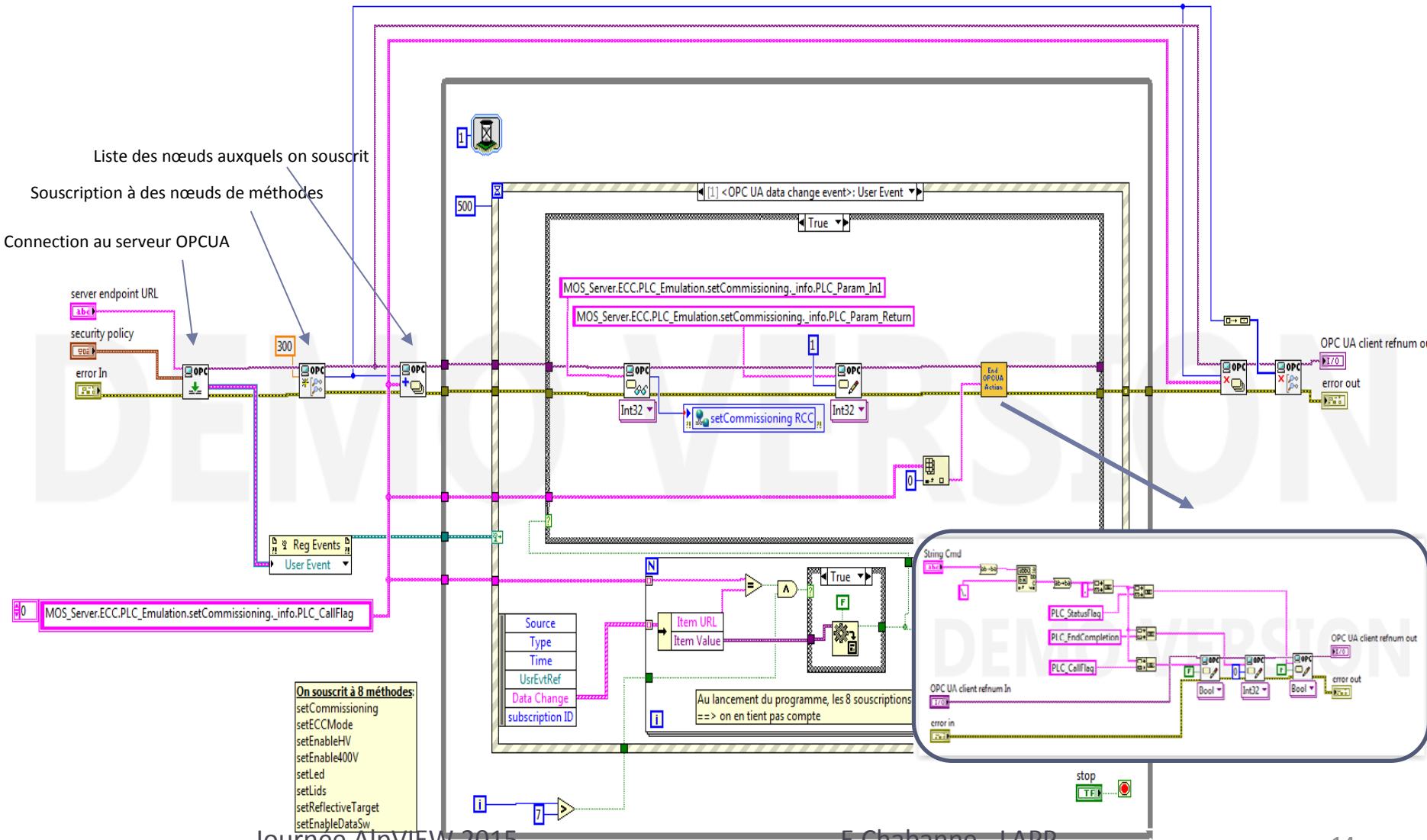
Lecture de la valeur de nœuds



Ecriture de la valeur de nœuds sur le serveur OPCUA



Souscription à un/des nœuds de méthodes





Merci de votre attention



Questions ?

Contact : eric.chabanne@lapp.in2p3.fr

User Program
Labview

Labview environment

User Program
C/C++

OPCUA general server
LAPP library

Plugins

NectarCam
librairie

UA extended
Unif.Autom library

UA stack
UnifiedAutomation
Unif.Autom library

UA stack
National instrumentation
library

Tested
Architecture

- X86 32 bits
- X86 64 bits
- Arm V7a

Linux

- Fedora
- SL 6
- Debian