

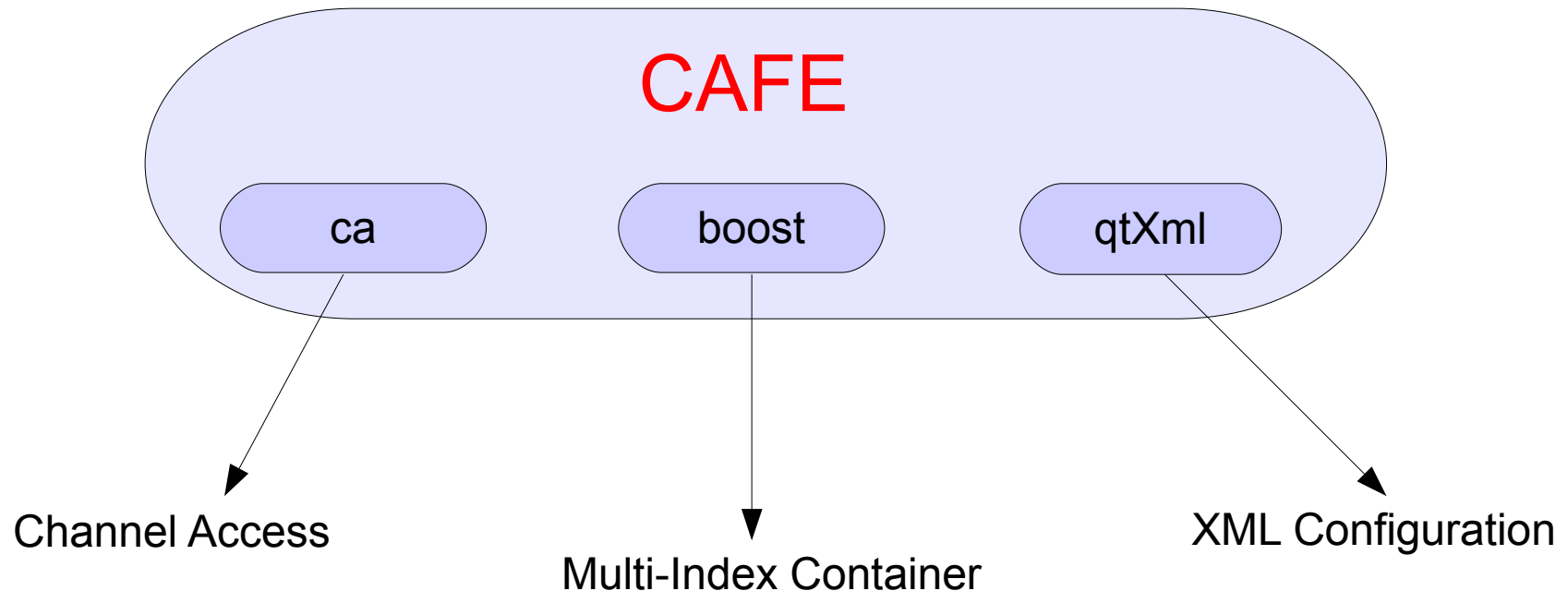


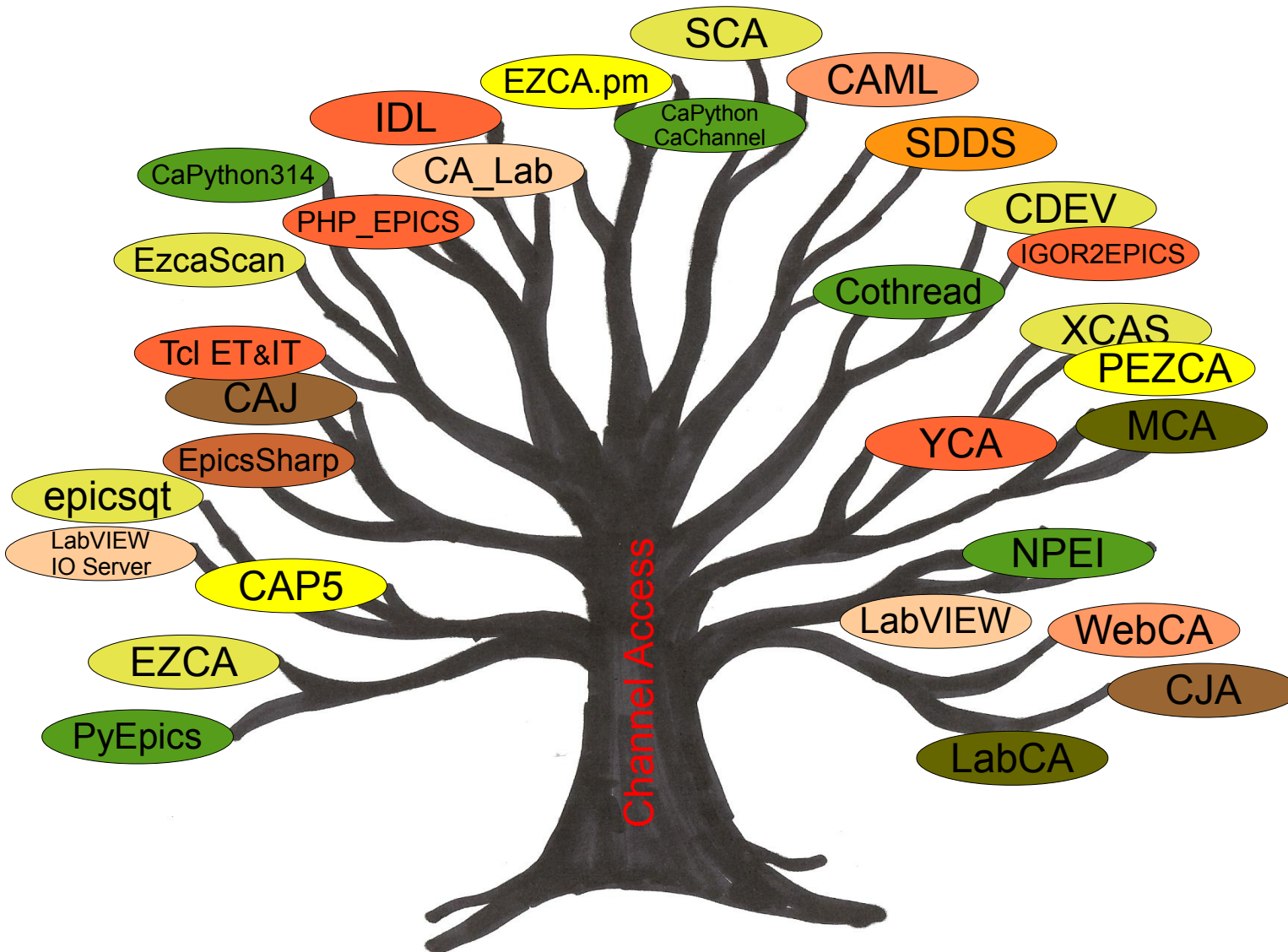
Hard Rock CAFE

Jan Chrin
Paul Scherrer Institut



CAFE, a C++ library





Advantages:

- Inherent simplicity and convenience of maintaining CA interface code (in a single place) and the flexibility it offers for designers
- New CA functionalities in future EPICS releases (A. Johnson, R. Lange @ this Meeting) can be readily integrated in a single location.
- Simplifies interface to declarative / 4th generation languages
- In-house CA client expertise ensures a quick response to user needs and problem solving

- New C++ library to replace CDEV API at SLS
Wishing to preserve device/attributes paradigm
notion of collections: group nodes of the same type
and views them as a single logical software object
Groups/collections easily configured with XML (cf. DDL for CDEV)
- Pulsed events at SwissFEL Injector Test Facility / SwissFEL
10 Hz: synchronous CA group invocations for machine DAQ
100 Hz: low-level DAQ required (B. Kalantari @ this Meeting)
- However, priorities changes as other user requirements surface
More extensive interface to MATLAB
Conclude: one C++ library (**CAFE**) as host API

Details:

- Synchronous, asynchronous interaction
for individual, collection/groups of channels (configured thru XML)
- Data passed through CA in native data type
data may be presented to client in any meaningful type
(true also of enumerated types)
- Extensive use of templates
(get/set operations, underlying CA classes, data conversions)

```
long CAFE::get(unsigned long _handle, float &_val)
```

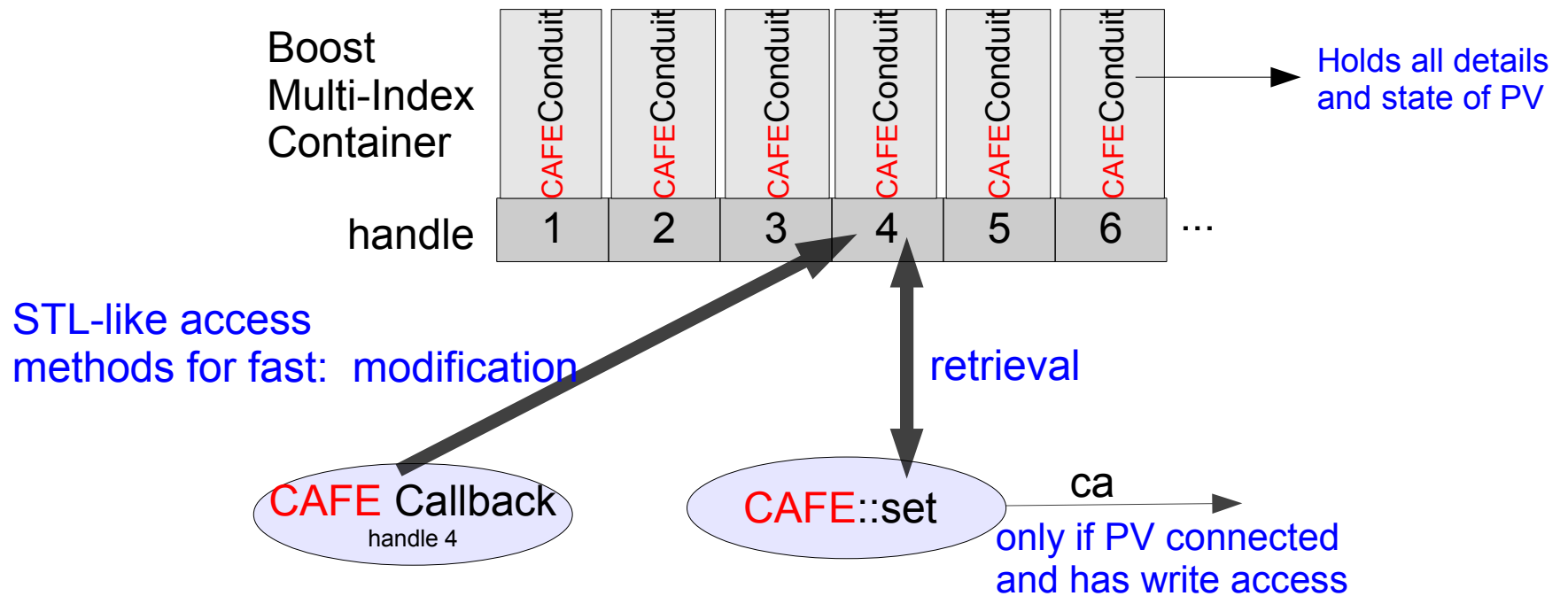
```
CAFEInstant<float> cafeFrappuccino;
```

```
status=cafeFrappuccino.get(_handle, DBR_FLOAT, &_val);
```

More Details:

CA callback functions implemented

for CA connection handlers, event handlers, access right handlers



Even More Details:

- STL-compatible access methods for fast retrieval and modification
e.g. `ca_state` inquiry ~100 faster than native CA client call
i.e. handle queries modifications are inexpensive
- All **CAFE** operations report their data/status to the Container
- Use of handles (object references) enhance performance
Dynamically (re-)allocate memory to store the handle's data in the Boost Container whenever CA connection handler invoked

Yet More Details:

- Hide low-level technical CA details

CAFE pre-configured with default CA parameters,
e.g. `long CAFE::get(unsigned long _handle, float &_val)`

is blocking without callback

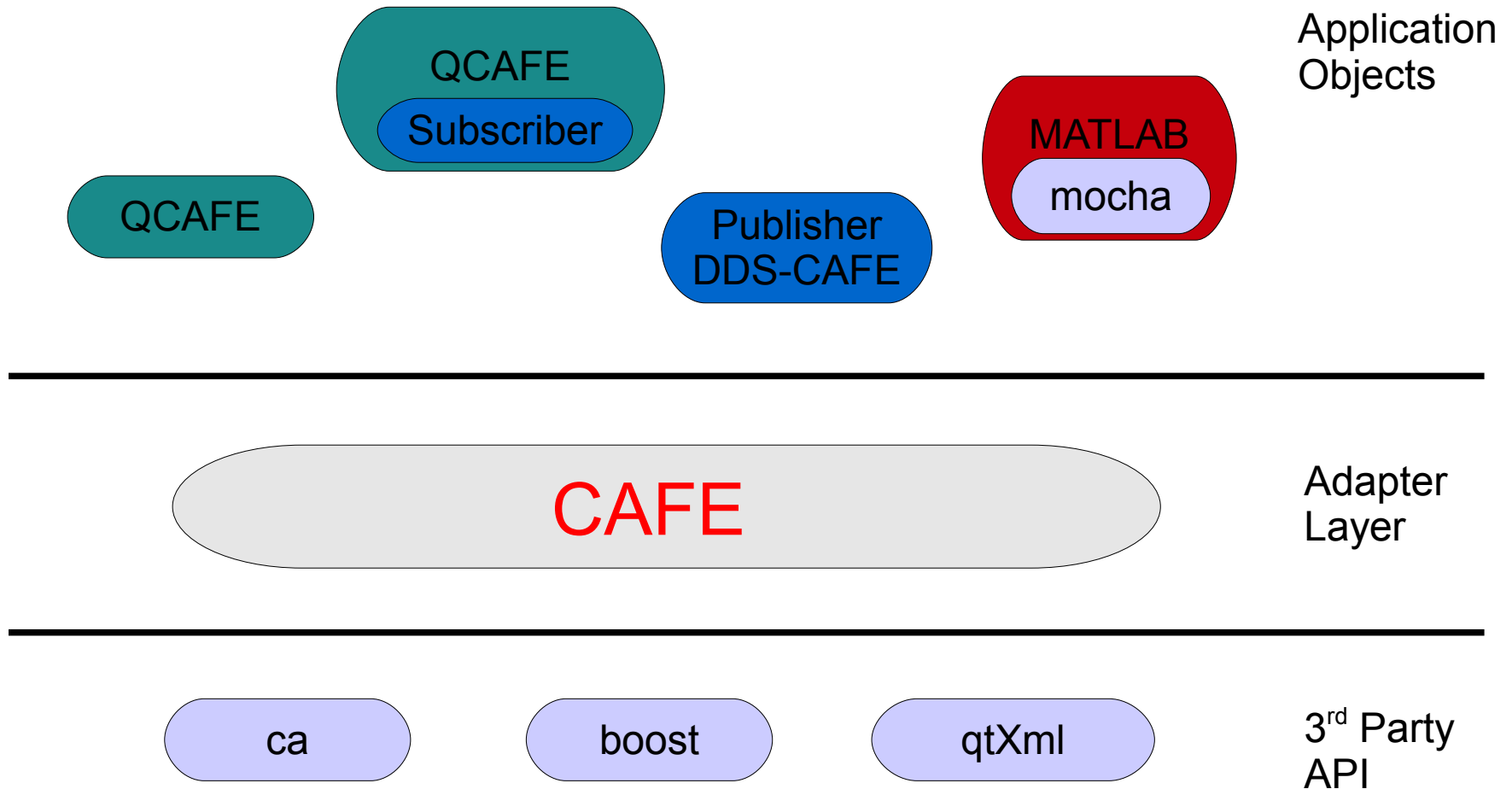
Handles possess their own CA properties, which can be configured
'on the fly' through designated container modification methods:

timeouts, blocking or non-blocking with callback

- CA Client Programming:

K. Evans, <http://www.aps.anl.gov/epics/docs/GSWE.php>

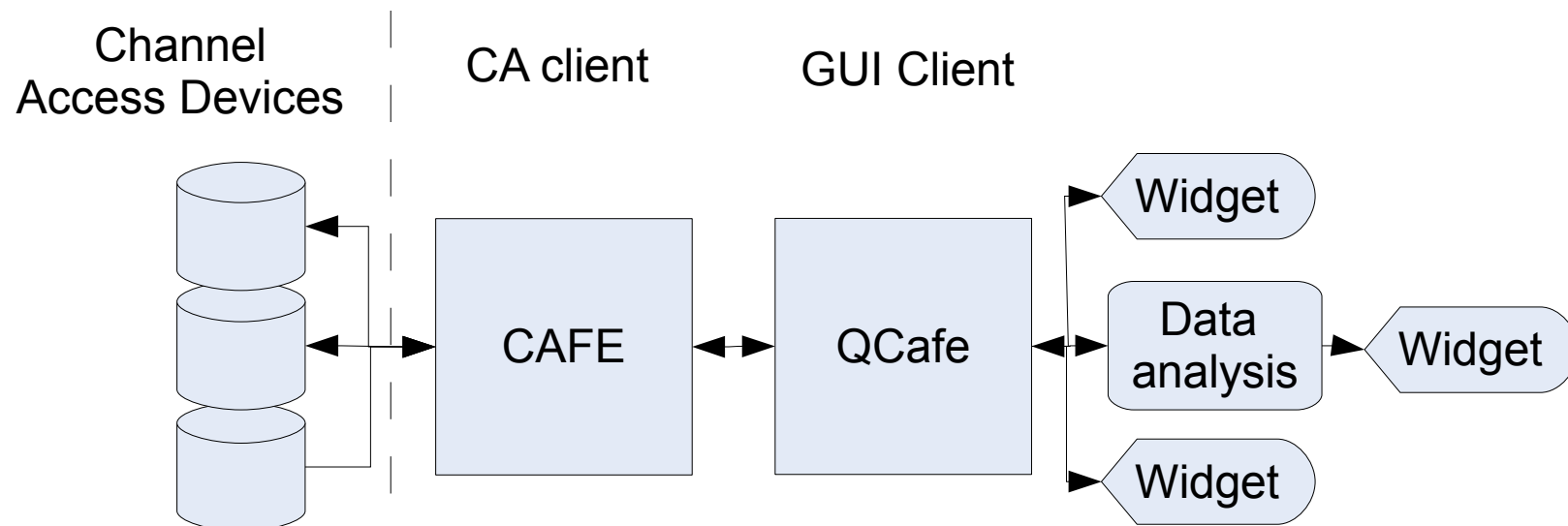
D. Zimoch, <http://www.aps.anl.gov/epics/meetings/2009-07/>



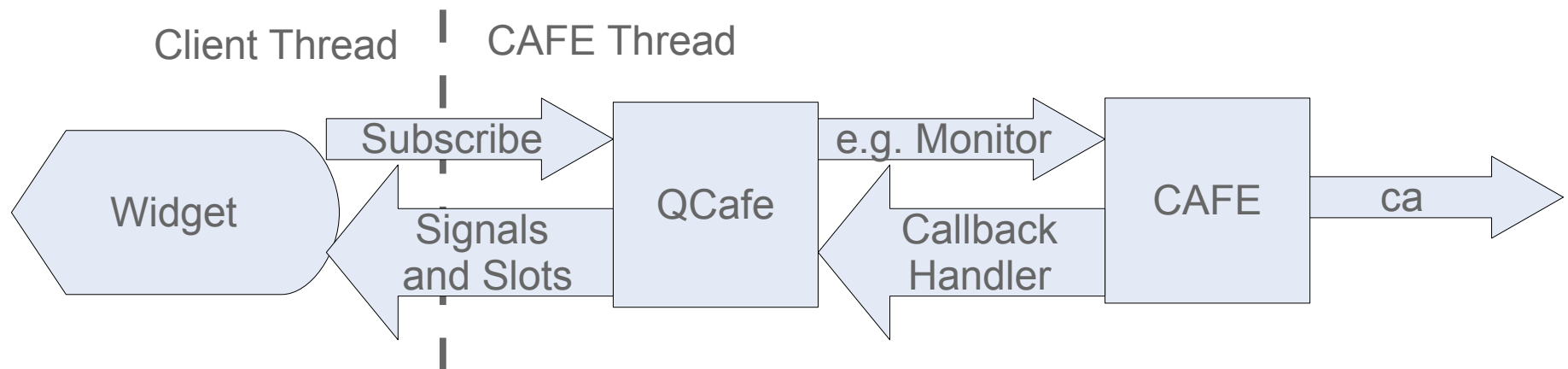
$$Qt + \text{CAFE} = \text{QCAFE}$$

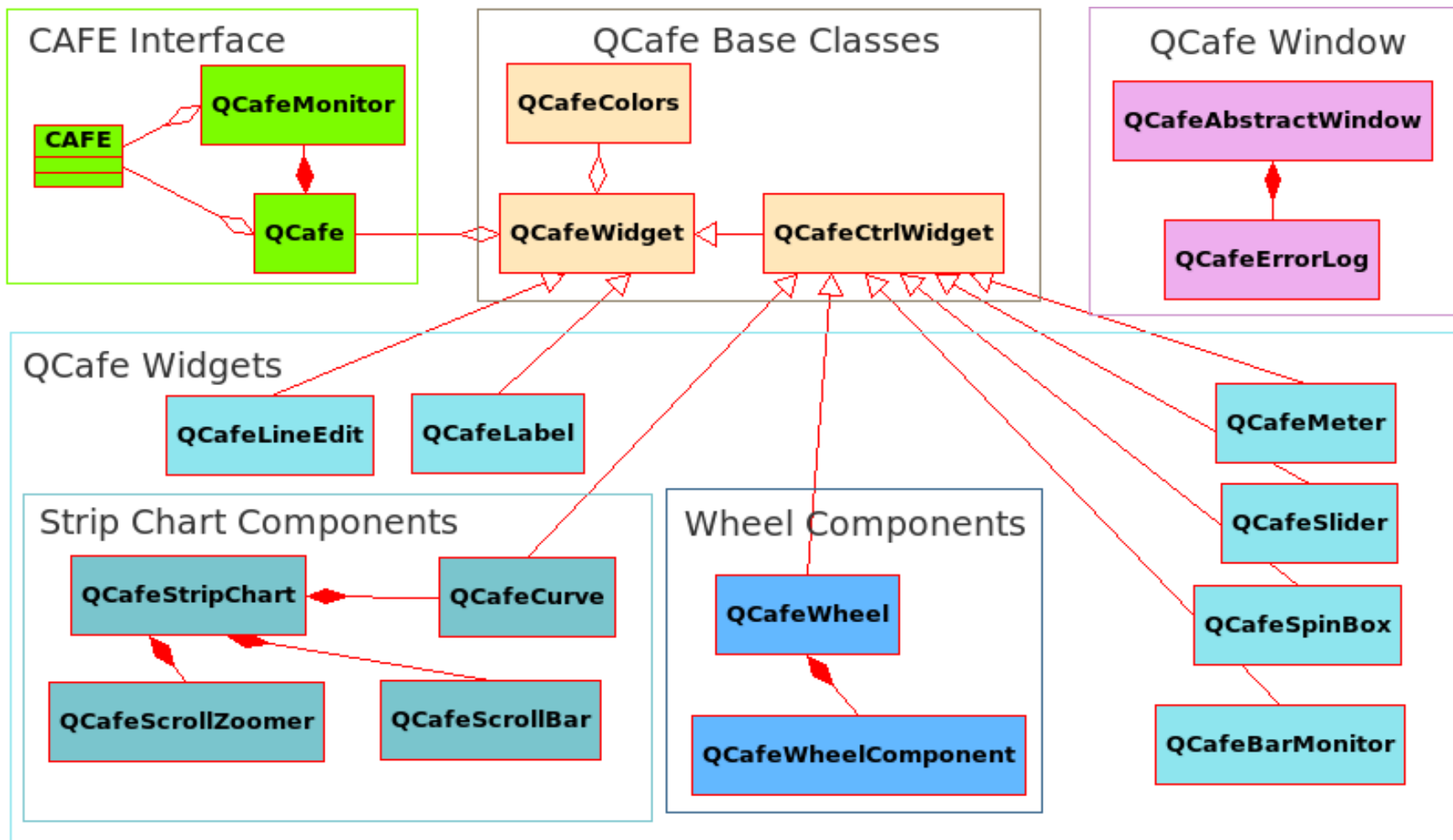
M.C. Sloan,
Principal developer,
Summer 2010

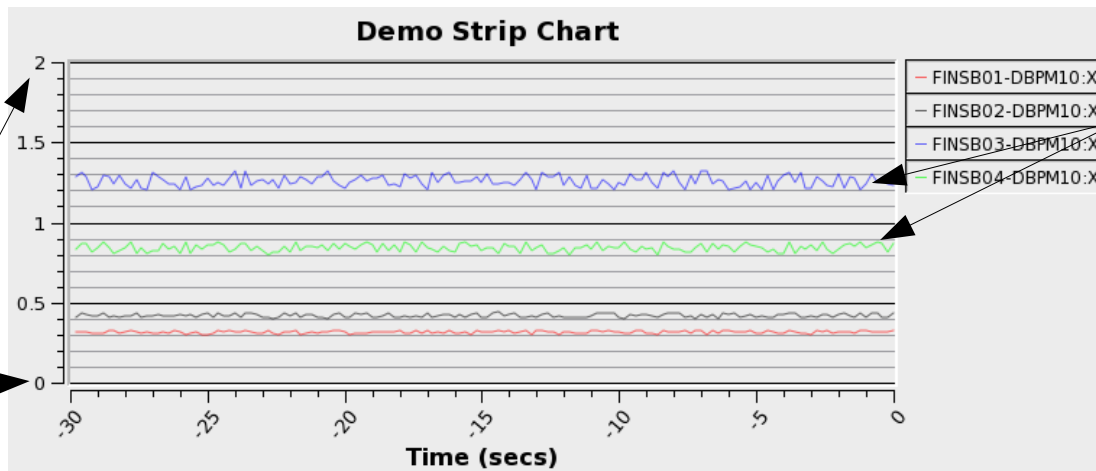
- Integration of Qt with CAFE
- Speed, flexibility and power of C++/Qt combined with independent and robust Channel Access using CAFE



- QCafe widgets decoupled from CAFE and channel access devices
- Data passed to widgets using Qt Signals and Slots
- Processor intensive analysis of data can occur in client thread

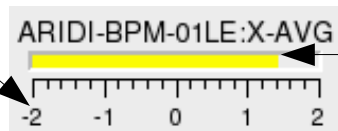






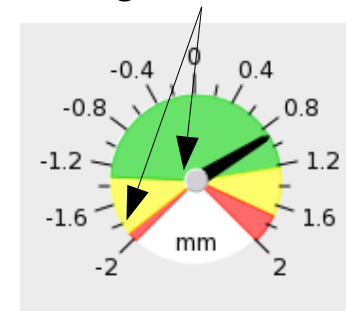
Real-time monitoring of multiple variables

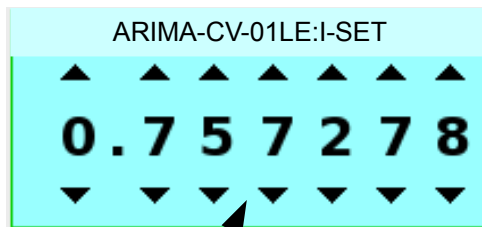
Automatic parameter configuration



Standardized colour warning indicators

Automatic configuration of warning and alarm limits



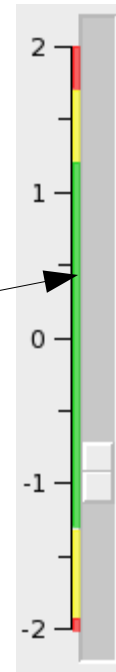
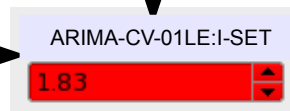


Modeled on MEDM wheel widget

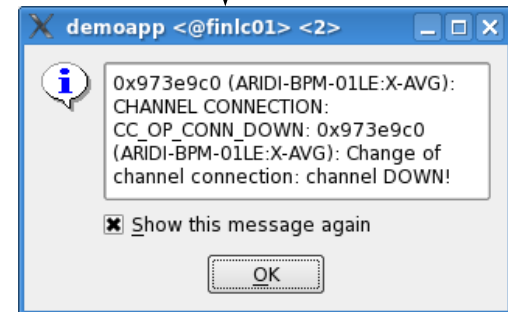
Adjusting the value automatically sets the EPICS variable

Wheel Widget: a complex widget composed of parts

Automatic colours and parameters similar to monitor widgets



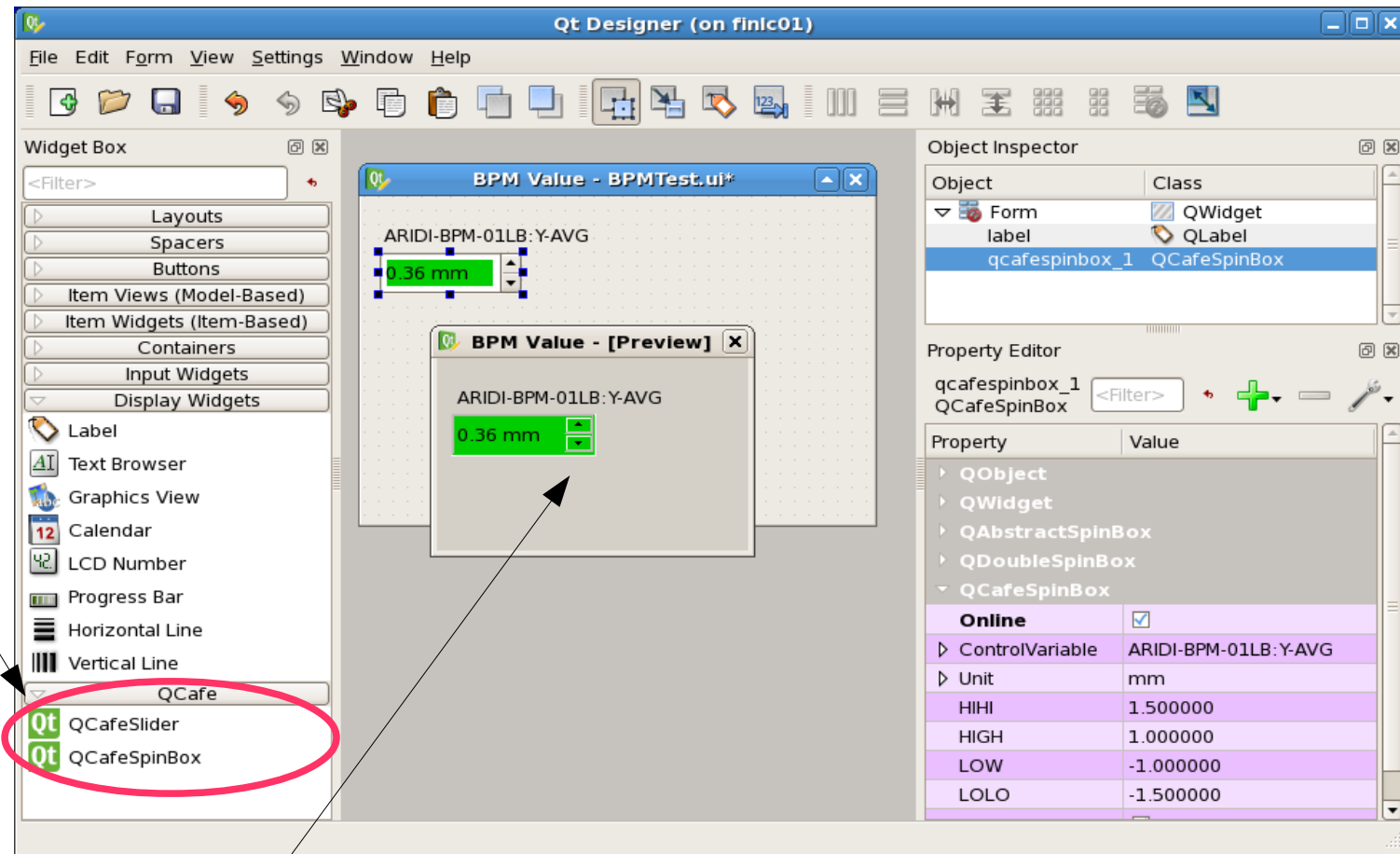
If an error occurs in a set, the user is informed and the value reset



Widgets turn to white when a disconnect occurs

R. Landstrom, Fall 2011

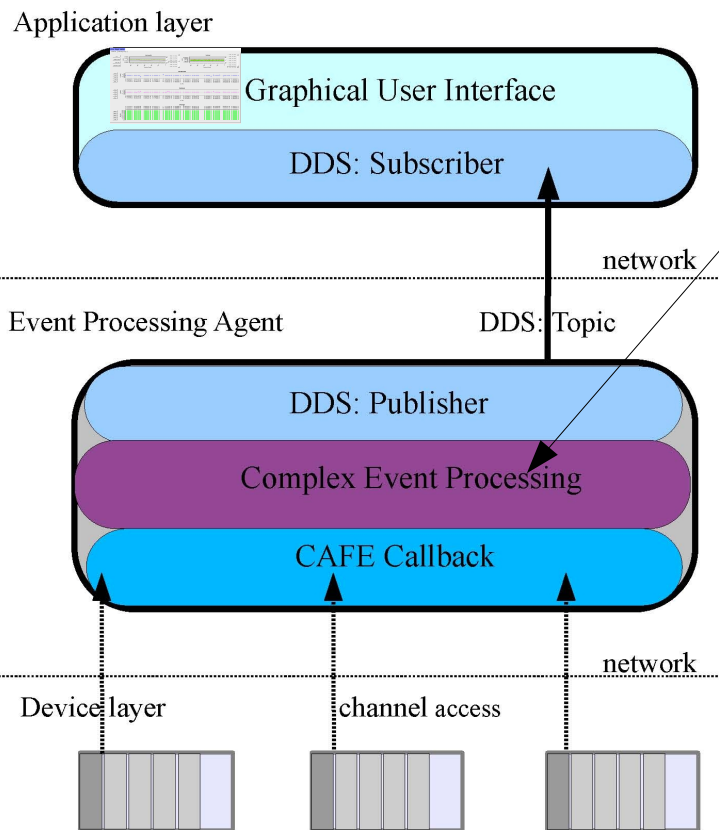
- GUI Builder
- QCAFE (epics-aware) widgets being added



Preview mode shortens development cycle

- J. Baek et al., “**KSTAR** Widget Toolkit using Qt Library for the EPICS Based Control System”, **ICALEPCS 2009**
- A. Rhyder, A. Owen, G. Jackson (**Australian Synchrotron**), “Qt EPICS Development Framework”, **PCaPAC 2010**
- E. Matias et al., “Phase II and III The Next Generation of **CLS** Beamline Control and Data Acquisition Systems”, **ICALEPCS 2011**

Data Distribution Service is a recent OMG standard: supports high performance publish-subscribe messaging with configurable Quality-of-Service (QoS) guarantees



- Event Processing Agents (EPAs) assigned task of aggregating, verifying, analyzing “group of related devices” e.g. Beam Positions Monitors (BPMs)
- Summarized results distributed by DDS: Publisher to interested clients DDS:Subscriber

groupName='gDBPM' defined in EPA XML-configuration file

```
long CAFE::openGroup(char * groupName,
                    unsigned long &grouphandle) throw (CAFEException);
```

```
long CAFE::monitorGroup (unsigned long groupHandle,
                        PVGroup &pvgroup, pCallback userCallbackHandlerMonitor);
```

./xmlFilter **SwissFEL.xml**
node=DBPM >
CAFECollection_SwissFEL.xml

```

<cafe:collection id="cDBPM">
  <cafe:description>Collection of DBPM SwissFEL
  Injector Test Facility </cafe:description>
  <cafe:member pos="0.4370">
    <cafe:device>FIND1-DBPM10</cafe:device>
  </cafe:member>
  <cafe:member pos="7.7130">
    <cafe:device>FIND100-DBPM10</cafe:device>
  </cafe:member>
  . . . . .
  <cafe:member pos="64.1155">
    <cafe:device>F10D100-DBPM10</cafe:device>
  </cafe:member>
</cafe:collection>
  
```

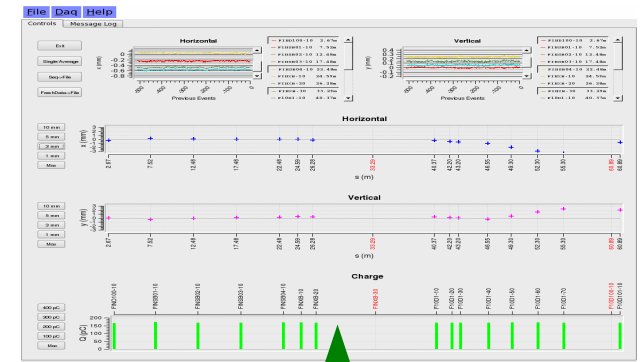
```

<cafe:group id="gDBPM">
  <cafe:description>DBPM EPA configuration
  </cafe:description>
  <cafe:collection> <cafe:id="cDBPM"/cafe:id>
    <cafe:attribute>X</cafe:attribute>
    <cafe:attribute>Y</cafe:attribute>
    <cafe:attribute>Q</cafe:attribute>
    <cafe:attribute>POS-ENABLE</cafe:attribute>
    <cafe:attribute>Q-ENABLE</cafe:attribute>
  </cafe:collection>
</cafe:group>
  
```

./xmlFilter **SLS.xml**
node=DBPM >
CAFECollection_SLS.xml

```

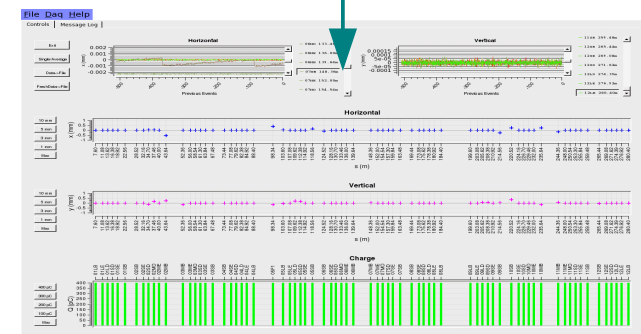
<cafe:collection id="cDBPM">
  <cafe:description>Collection of DBPM SLS
  </cafe:description>
  <cafe:member pos="11.0775">
    <cafe:device>ARIDI-BPM-01LE</cafe:device>
  </cafe:member>
  <cafe:member pos="11.6225">
    <cafe:device>ARIDI-BPM-01LD</cafe:device>
  </cafe:member>
  . . . . .
  <cafe:member pos="280.3975">
    <cafe:device>ARIDI-BPM-12LB</cafe:device>
  </cafe:member>
</cafe:collection>
  
```



SwissFEL Injector Test Facility

SwissFEL EPA

SLS EPA



SLS Storage Ring

MATLAB Objects for Channel Access

mocha MEX-file interfaces **CAFE** to MATLAB (**mocha/mca** single MEX-file)

cf. labCA interfaces EZCA to MATLAB/SCILAB --> several MEX-files

sizeof (**mocha code**) << sizeof (mca code)

Binding procedure follows a recognizable pattern:

- Validate input arguments
- Invoke **CAFE** method
- Convert **CAFE** output arguments into MATLAB data type
- Prepare data format (array, struct..) for return to caller

mocha mca-compatible MATLAB scripts provided to facilitate transition from mca to **mocha**

MATLAB Objects for Channel Access

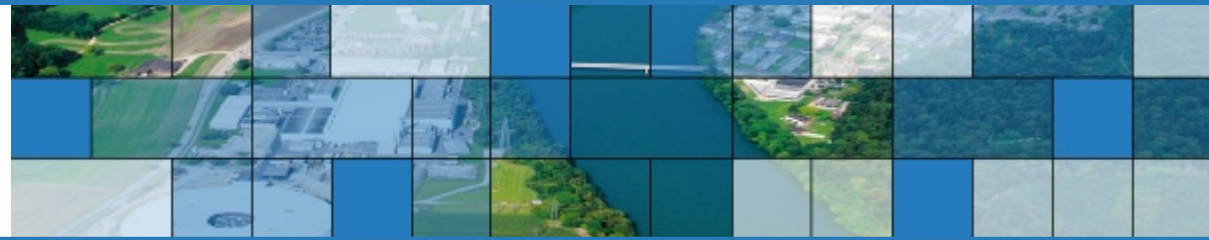
mocha:

- Access to all MATLAB primitive data types (analysis of camera data)
- MATLAB oo: mocha-aware classes (orbit data, machine snapshot)
- classes, on initialization, extract static data, i.e. node names, positions from the accelerator XML (computation moved to mocha MEX-file)
- Compilation on a 64-bit Linux server (machine data / 3D simulations)

mocha monitors:

- Cache updated values, optionally execute a script to update widget (but MATLAB's Event Listener model would be a better approach)

- **CAFE**, ingredients for a credible, 'in-house' C++ CA API, sufficiently modular, expressive and flexible to be applied in other domains
 - Q**CAFE** (+plugins to Qt Designer) high-performance GUI applications
 - CAFE**-DDS state-of-the-art publish-subscribe for reactive programming
 - mocha** MEX-file demonstrates bindings to scripting languages are simplified by using **CAFE** as a host; tangible benefits for MATLAB users
- Consolidate code, first large-scale deployment at SLS (as replacement to CDEV)
- Attract users (to test and make better).
- Visit us @ ICALEPCS 2011, Poster ID: WEPKS024



ALL IS ONE

LOVE ALL – SERVE ALL