Minutes of LHC-CP Link Meeting 19

Subject	:	LHC Controls Project
Date	:	920 November, 2001
Place	:	936-R-030

Participating Groups :

EST-ISS	apologies,
LHC-ACR	Ph. Gayet,
LHC-ECR	no representative,
LHC-IAS	no representative,
LHC-ICP	apologies,
LHC-MMS	no representative,
LHC-MTA	no representative,
LHC-VAC	I. Laugier,
PS-CO	F. DiMaio,
SL-AP	E. Wildner,
SL-BI	no representative,
SL-BT	E. Carlier,
SL-CO	A. Bland,
SL-HRF	L. Arnaudon replacing E. Ciapala,
SL-MR	R. Billen,
SL-MS	no representative,
SL-OP	M. Lamont,
SL-PO	S. Page replacing Q. King,
ST-MO	L. Scibile replacing P. Solander.

Others :
P. Charrue (SL-CO),
C. Frisk (Workstation Support)
R. Hopkins (Workstation Support),
K. Kostro (CMW),
R. Lauckner (Chair),
M. Peryt (Logging Project)
B. Puccio (Machine Protection),
G. Segura (RAMSES)
M. Tyrrell (Alarm Project),
M. Vanden Eynden (Core Team).

Distribution :
Via LHC-CP website: **http://cern.ch/lhc-cp**
Notification via: **lhc-cp-info@cern.ch**

Agenda :

1. Matters arising from Previous Meeting	
2. LHC-CP News	R. Lauckner
3. Back Ends for LHC	R. Hopkins, P. Charrue
4. RAMSES	L. Scibile, G. Segura
5. Project Middleware	S. Page, K. Kostro
6. AOB	

1. Matters arising from Previous Meeting

The urgent actions from October 9th concerning fieldbus cables and ergonomics for the vacuum supervision have not yet been followed up.

ACTION: P. GAYET

R. Lauckner is following the request for underground control rooms from the same meeting – this moves to the long-term actions.

2. LHC-CP News R. Lauckner

The QRL planning has been base lined and the first monthly tracking meeting took place on 16th November. Apart from verifying progress 3 new technical questions were raised

- An Ethernet WorldFIP bridge is required from LHC-IAS
- A policy for displaying calendar time and converting from UTC is required.
- Support for the Cryogenic Instruments calibration database is requested.

These issues will be followed up at future tracking meetings.

The sector test planning is being compiled. The last input was from the SL-BI group and SL-BT will be next.

Recent developments in SPS division concerning the LTI Project, the CNGS project and the SPS2001 project have convinced the division that these projects must be managed coherently with the LHC work. A first step is that M. Lamont has taken over as SPS2001 project leader reporting to the LHC-CP.

R. Lauckner also reported on the Controls Working Group that has been asked by the CERN re-structuring Task Force 5 to address the controls issue in the accelerator sector. The group consists of P. Charrue, B. Frammery, R. Lauckner, P. Ninin and C-H Sicard. He explained the views of the Core Team on this issue.

M. Vanden Eynden reported that the Timing Functional Specification is in the EDMS approval system.

The schedule and main topics for the next LHC-CP meetings are:

4/12	Status of Real Time Controls, Operational data requirements	Wijnands, Billen.
18/12	Future Front Ends, Requirements for Analogue Signals	Vanden Eynden, Ciapala

3. Back Ends for the LHC R. Hopkins

R. Hopkins explained that the term Back Ends refers to the High Availability Application

and File Servers for the LHC Controls System. He explained the current situation in the PCR. There are parallel operation and development environments each equipped with application and fileservers. The operational environment also has a display server to support continuously running programs generating permanent status displays while the development environment has a development server. Recent evolution has been the relocation of all personal accounts in the development environment to an IT supported server. SL supports operational accounts. However he emphasised the importance of direct support from the server vendor for critical machines.

Another recent evolution in the PCR has been the introduction of Windows 2000 machines as Java GUI platforms. He pointed out that this is not a back end machine. They are not intended to support accelerator 'business' applications.

R. Hopkins went on to elaborate important considerations for determining the future evolution of the back ends. Faced with severe reduction in CERN manpower workstation support was his main concern. He favours a combination of a CPU Server and a Fileserver. The powerful multi-CPU server would be capable of hosting HPUX, Linux and NT applications. The Fileserver would also provide Front End support.

F. DiMaio considered that this was an expensive solution. In the PS the control room functionality is provided by local Linux PCs and fileservers, the latter managed by the IT division. Vendor maintenance contracts had been replaced by a 3 year guarantee attached to the initial purchase of the PCs. R. Hopkins confirmed that SL database servers are managed by IT and pointed out that his proposal is tailored to providing stable mission critical platforms to support the evolving modern software architecture increasingly employed at SL.

R. Lauckner raised the question of support for supervision software and platforms for industrial computing. After some discussion it was agreed that this question should be studied in the context of control systems involved in the QRL reception tests in 2003.

ACTION: A. Daneels, R. Lauckner

4. RAMSES Project L. Scibile, G. Segura

L. Scibile and G. Segura presented the Radiation Monitoring System for the Environment and Safety (RAMSES) project. This project will produce the next generation of radiation monitoring equipment to succeed the LEP era ARCON system. The scope includes the site monitors, local alarm panels, forwarding alarms to the CERN Alarm System and providing an historical archive. The LHC machine and experiments will be covered in the first phase. Later it is expected to replace the ARCON system CERN wide. TIS are the main users but the service will also be provided to accelerator and experiments control rooms.

The project team is lead by L. Scibile from ST-MA and includes 4 project engineers from TIS. The first system is to be operational for the Sector Test in 2004.

High reliability is required for the monitoring and capture of the radiation dose information. This functionality must be guaranteed independently of ancillary systems. The archival will require a database service, this will probably be submitted to INB controls.

The project is in the market survey phase and in parallel L. Scibile explained that they are collecting user requirements. He asked all LHC-CP members to contact him if they have requirements.

ACTION: All LINKMEN

A preliminary list had been established at the last Controls Board meeting. This is reproduced here:

1. Machine Protection System Link person: R. Schmidt
2. Post Mortem Facility Link person: J. Wenninger
3. Timing System Link person: G. Beetham
4. Alarm System Link person: M. Tyrrell, U. Epting
5. Logging System Link person: R. Billen, M. Peryt

P. Gayet asked why this facility could not be treated as an extension of the CSAM. L. Scibile replied that there are indeed similarities especially in the upper layers of the architecture. At the moment this issue is completely open, another possibility might be an off the shelf solution.

R. Billen commented he did not see the hosting of the database as a critical issue. It will be necessary to link the operational database(s) with the radiation database.

5. Project Middleware S. Page, K. Kostro

S. Page explained the Control System architecture for Power Converter control. Around 1800 converters will be connected via WorldFIP links through Front End gateways to a central application server. There will be a maximum of around 30 converters on a FIP segment and about 100 gateways. Communication via the gateway server and the application client will employ the Common Middleware, CMW. A Java Class in the client will implement all the services required by the Power Converter applications.

The currents for LHC ramping depend upon the history of the injection process. Consequently ramping functions must be calculated and loaded in a large number of converters with a minimum time delay to optimise machine performance. A large number of instructions must be sent to a large number of devices and there will be no time to wait for responses between instructions.

S. Page has tested a Thread Pool server side middleware that assigns one thread per set or get request. This allows asynchronous calls to be processed concurrently. He reported on tests showing that the handling of the number of gets submitted per second scaled very well with the number of threads in the server and the number of client threads. Also the number of server responses received per second scaled well with the same parameters.

These tests were run on a single machine and on a pair of networked machines. The performances were better in the second case indicating that the Java client had been the limitation to increasing the number of clients with a single machine.

This test has shown that *thread per request* would be a viable possibility to support non-blocking requests in the server. However the next version of CMW will provide asynchronous request support in the server, offering even better performance and better adapted to the current CMW implementation, which is based on ORBacus/E.

S. Page clarified the proposed control system architecture pointing out that the focus was on get/set performance and the real time client support indicated was preliminary.

K. Kostro reported on OPC support in the CMW. He explained that backed as an industry standard by Microsoft OPC servers are now widely available for connecting hardware devices and communicating with SCADA systems. A server side OPC gateway is now provided as a component of the CMW, this implements get/set and subscribe services on the client side for OPC connected devices. He reported on tests with 3 different OPC servers, Siemens Win CC (BT targets), PCVue (String 2) and Schneider PLC (RF Focus Power Supply). He emphasised that the CMW OPC gateway used in the 3 cases was exactly the same and configured from an Oracle based configuration service.

With a Device Explorer developed by the CMW team he demonstrated how an Oracle based naming service can be used to explore these devices down through their class, device and property layers. The demonstration of access to OPC servers carried out during his presentation included updating RF Focus voltage values after switching power on and reading of String 2 alarm values. Accesses to PS Booster equipment (using PS equipment server on LynxOS) and to SPS SEM (using SL-Equip gateway on Linux) via the same tool were also shown.

P.Gayet reported that the next version of PVSS would have an OPC server

Marc Vanden Eynden announced that SL Front Ends will be upgraded to LynxOS 3.1.0A during the present shutdown which will allow them to support the CMW Middleware server.

F. DiMaio reported that the AD machine had successfully switched to equipment access through CMW for the last 2 weeks of the physics run.

6. AOB

There was no further business.

Long-Term Actions	**People**
Underground Control Rooms requested	R. Lauckner
Establish Post Mortem sub-project	R. Lauckner
Clarify Middleware Services to be used by LHC-CP	Core Team

Reported by R. Lauckner

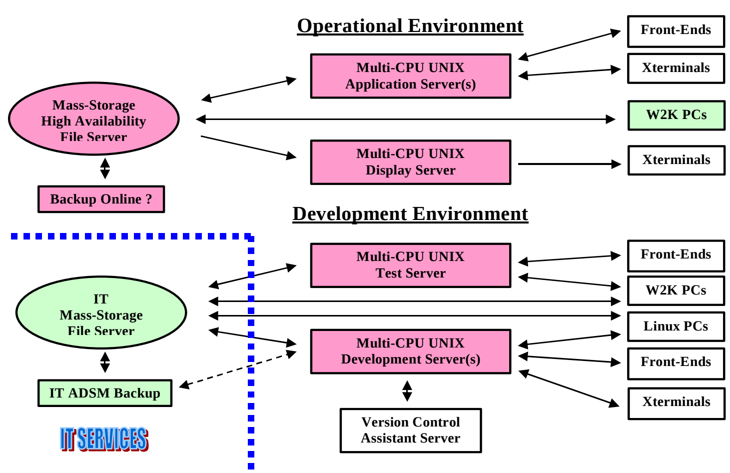
LHC-CP Backends

R Hopkins (SL/CO/WS)
20th November 2001

LHC-CP Backends

High Availability Application
and File Servers for the
LHC Controls System

The Situation Today



Important Considerations - 1

- Reduction in CERN manpower.
  - "High availability" hardware solution.
  - Minimum manpower support solution.
- Flexible hardware solution.
  - Platform giving different OS options.
  - Platform giving **scalability** into the future and beyond the original design criteria.

Important Considerations - 2

- Solution with a clear upgrade policy.
  - Maintain hardware equality with techno-evolution using an existing basic hardware framework.
- Consider a vendor offering a lease-plan.
  - Assist purchase (smooth budgeting).
  - Provide auto-upgrade strategy.
  - Ensure guarantee and support.

Important Considerations - 3

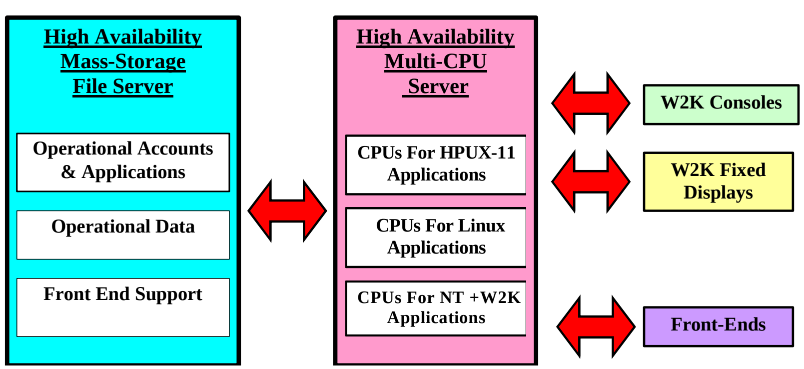
- Hardware Support.
  - Proprietary solution from a major vendor.
  - Contractual support from that vendor.
- Question :

  Why should proprietary contractual support differ from an outsourced support contract favoured by CERN ?

Proposed Solution

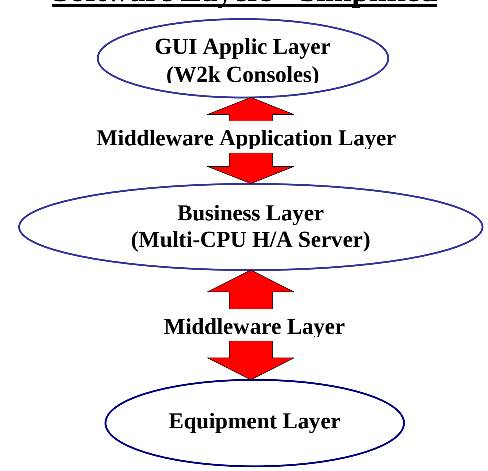
Topology Proposal For SPS & LHC Servers

"Mission-Critical" Operational & Development Environment



Does The Proposed Hardware Solution Fit The Software Requirements ?

Software Layers - Simplified



General Comment

- The requirement of File Servers and Application Servers as a backend hardware solution for LHC should be considered “Mission Critical”.
- The hardware support contract with the supplier of these servers should be considered as an out-source contract.

Conclusion

- Our mission is to provide LHC Operation with High Availability File and Application Server capability.
- We consider that a fully supported centralised backend solution will offer the best service.

RAdiation Monitoring System for the Environment and Safety (RAMSES)

The RAMSES Team

Presented by
L. SCIBILE (ST/MO) and G. SEGURA (TIS/TE)

RAMSES Project

Mandate / Scope

- The RAMSES will provide CERN with an integrated radiation monitoring system for the Environment and Safety covering acquisition, transmission, logging and display for the LHC machine, LHC experiments and experimental areas.
- TIS will exploit this system to assess radiation risks and to control the release of radioactivity. The system will also be incorporated into the accelerator and experiment control rooms.
- The mandate of the project team covers the system specification, prototyping, tendering, installation and integration of radiation monitors and industrial control equipment for safety systems.

RAMSES Project

Organisation

The project organisation will be as follow:

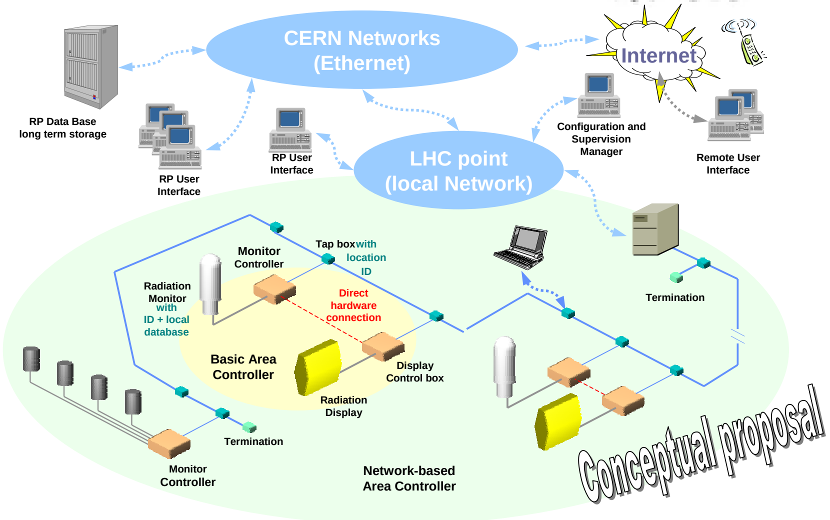
Product Owner	L. Evans	DG (for LHC)
(Ma’tre d'Ouvrage):	H. Schšnbacher	TIS (for other CERN sites)
Product Manager	P. Faugeras	AC (for LHC)
(Ma’tre d'Ouvrage D lgu)	H. Menzel	TIS (for other CERN sites)
Project Owner (Ma’tre d'Oeuvre):	H. Menzel	TIS
Project Leader (Ma’tre d'Oeuvre D lgu)	L. Scibile	ST/MO
Project Engineers "*Radiation Protection*"	D. Folker-Wirth	TIS/RP
Project Engineers "*Environmental Safety*"	P. Vojtyla	TIS/TE
Project Engineers "*Radiation Monitors*"	D. Perrin	TIS/RP
Project Engineers "*Software Infrastructure*"	G. Segura, L. Scibile	TIS/TE and ST/MO

RAMSES project

Planning

- Project Launch 10/10/2001
- Specifications May 2002
- Contract(s) from late-2002 -> early-2003
- First operational monitored area for injection tests 2004
- Complete LHC installation by 1st quarter 2006

RAMSES Project



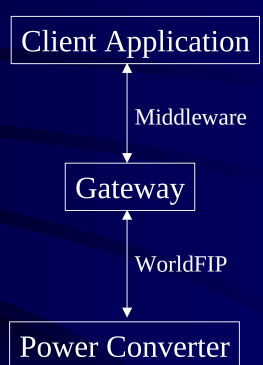
Prototype Power Converter Control System

Stephen Page

Presentation Format

- In two parts:
  - – Description of control system design
    - Set & get operations
    - Publish / subscribe operations
  - – Results of performance tests of ‘thread-pool’ middleware

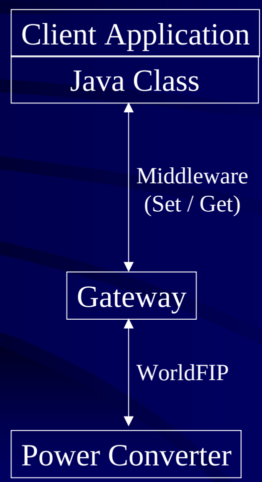
Basic System



Set & Get Operations

- Client interface is provided by automatically generated Java class
- Class handles interaction with middleware
- Set / get request arrives on the gateway
- Request is processed by a ‘binding layer’ before being passed to the gateway server
- Gateway relays request to converter

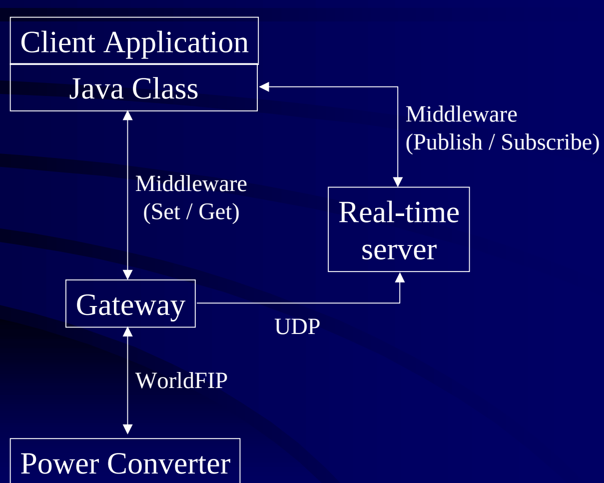
Control System Architecture



Publish / Subscribe Operations

- Publish / Subscribe methods will be integrated into the Java class
- Real-time server receives data from all gateways via UDP
- Real-time server handles subscription requests from clients

Control System Architecture



Performance Tests of Thread Pool Middleware

Requirements

- During preparation for ramping LHC many (approx. 6000) instructions will need to be sent to around 1800 devices situated around the LHC
- Instructions must be sent within a short period of time (~ 5 seconds)
- No time to wait for responses between instructions

Description of Thread Pool Middleware

- Allows one thread per instruction (set or get) on the server side of the middleware
- Non-blocking (asynchronous) instructions may therefore be processed concurrently
- Threads are taken from a pool (not created on-the-fly) to limit overhead

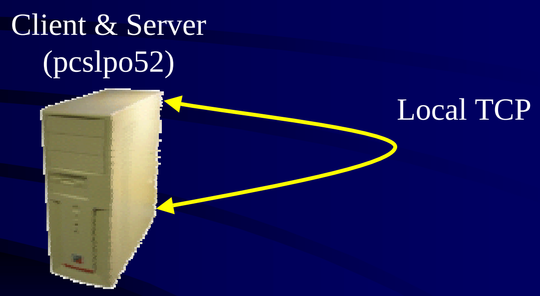
Test Server (C+ +) Description

- Delivers arbitrary value (current time) when get request received
- Sleeps such that each get request takes 1 second to process
- Variable size of thread pool

Test Client (Java) Description

- Sends non-blocking (asynchronous) get requests to server through middleware at fastest possible rate
- Variable number of client threads with independent connections to the middleware server

Test 1 - Local Client & Server - Setup



Test 1 - Local Client & Server - Machine Configuration

- Processor: 1x Pentium III 850Mhz
- RAM: 128Mb
- OS: RedHat Linux 7.1
- JDK: Sun 1.3.1

Test 1 - Number of gets submitted per second

Number of client threads

Threads in server pool

	1	10	50	100
1	3979.2	3596.0	4019.6	FAILED
10	3406.4	3559.6	3202.0	FAILED
50	3380.0	4011.0	3168.6	FAILED
100	3094.0	3301.2	2990.8	FAILED

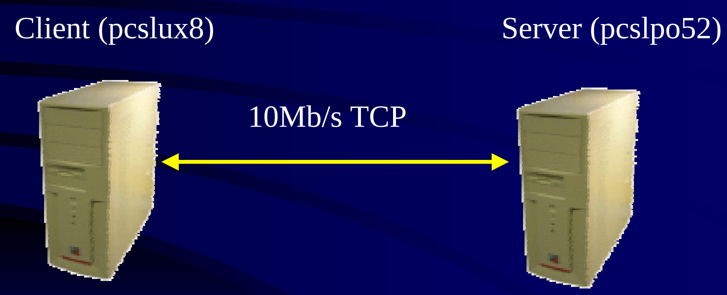
Test 1 - Number of responses received per second

Number of client threads

Threads in server pool

	1	10	50	100
1	1.0	1.0	0.8	FAILED
10	8.2	10.0	10.0	FAILED
50	48.0	50.0	42.6	FAILED
100	100.0	108.8	102.4	FAILED

Test 2 - Remote Client & Server - Setup



Test 2 - Remote Client & Server - Machine Configuration

Client

- Processor: 2xPentium III 600Mhz
- RAM: 256Mb
- OS: RedHat Linux 6.1
- JDK: Sun 1.3.1

Server

- Processor: 1xPentium III 850Mhz
- RAM: 128Mb
- OS: RedHat Linux 7.1

Test 2 - Number of gets submitted per second

Number of client threads

Threads in server pool

	1	10	50	100
1	4348.0	3950.6	3170.6	3197.0
10	3959.0	3769.8	3353.6	2964.4
50	4093.4	3652.6	3166.8	3172.8
100	3641.6	3784.2	3426.0	2959.0

	1	10	50	100
1	3979.2	3596.0	4019.6	FAILED
10	3406.4	3559.6	3202.0	FAILED
50	3380.0	4011.0	3168.6	FAILED
100	3094.0	3301.2	2990.8	FAILED

Test 2 - Number of responses received per second

Number of client threads

Threads in server pool

	1	10	50	100
1	1.2	1.0	1.0	1.2
10	12.0	10.0	10.0	10.0
50	50.6	50.0	50.0	60.0
100	103.8	100.0	115.4	100.0

	1	10	50	100
1	1.0	1.0	0.8	FAILED
10	8.2	10.0	10.0	FAILED
50	48.0	50.0	42.6	FAILED
100	100.0	108.8	102.4	FAILED

Conclusions

- Threads in pool scale well and rate of responses is directly proportional to number of threads
- With small data items, the network does not seem to add a large overhead.
- An increase in the number of clients does not reduce the responsiveness of the system significantly.

Future Tests

- Thread-per-client version of Controls Middleware based upon Orbacus-e
- Bulk publication through new Controls Middleware implementation

OPC Support in CMW

Kris Kostro

Outline

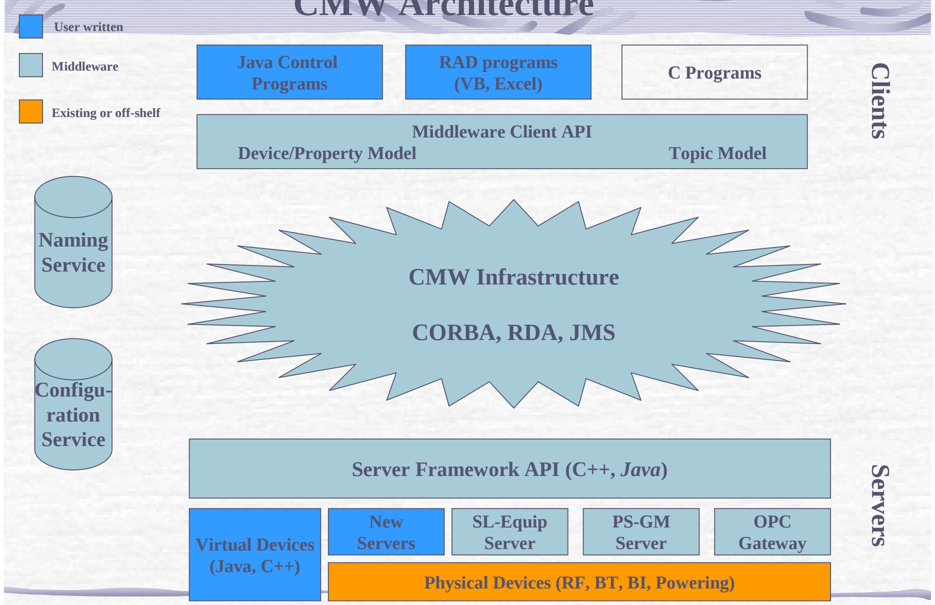
- Explain OPC
- Explain CMW Architecture
- How OPC is integrated in CMW
- Experience until now
- Demonstrate OPC access

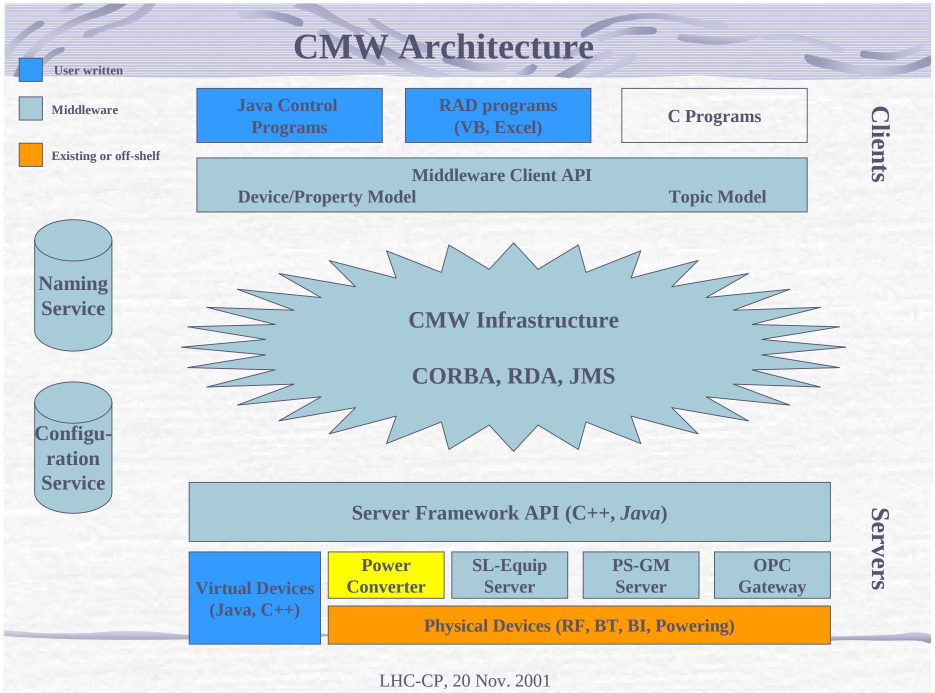
What is OPC

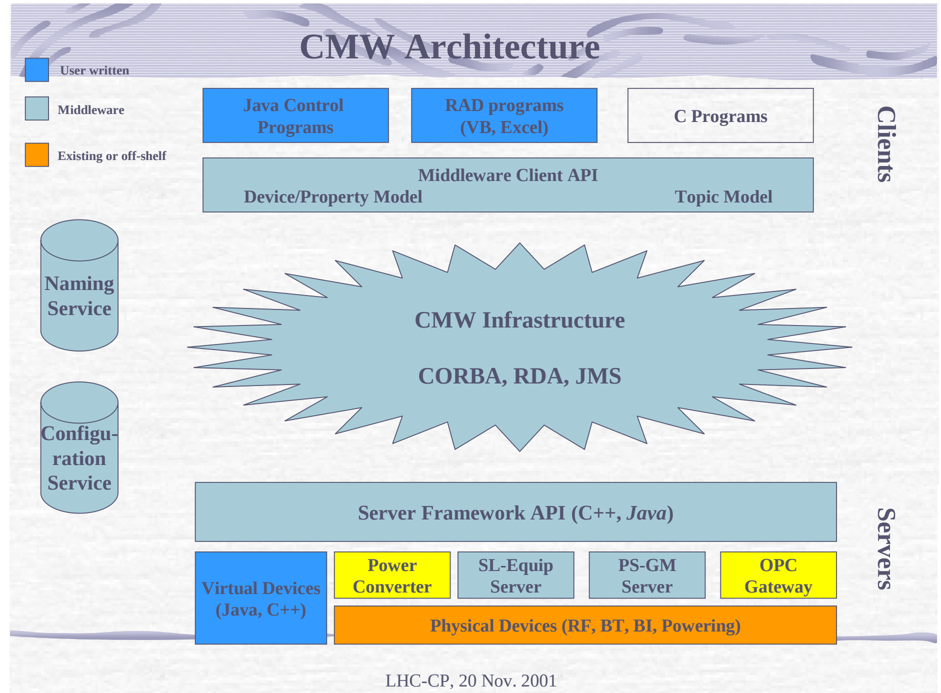
- Industry standard endorsed by Microsoft
- Based on COM (OLE for Process Control)
- OPC servers delivered with HW (like drivers)
- Widely used by SCADA
- Often SCADA Systems offer OPC Server to access SCADA RT DB

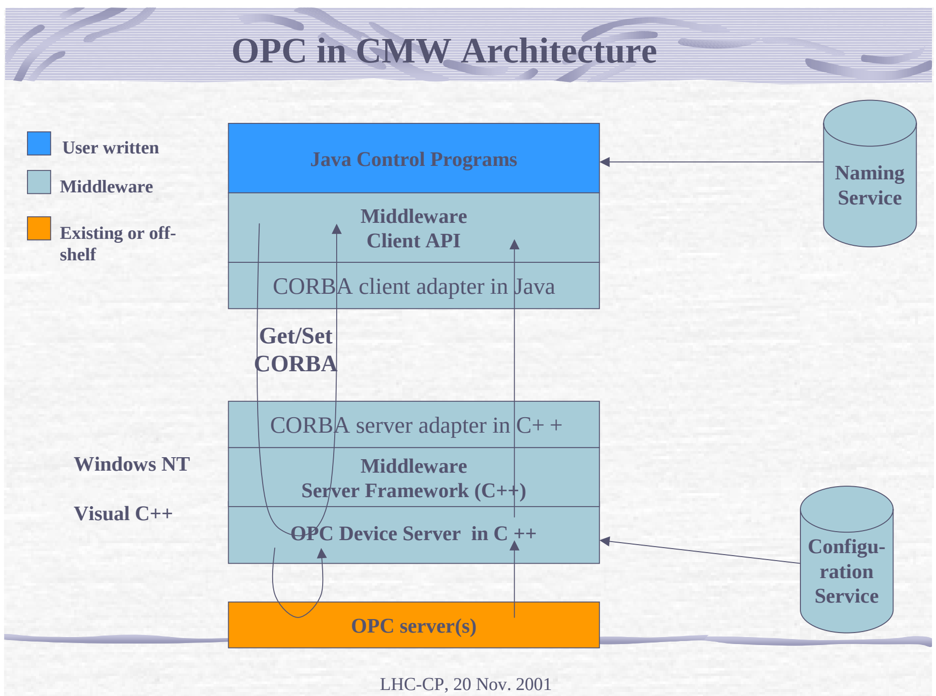
http://proj-cmw.web.cern.ch/proj-cmw/technology/opc.htm

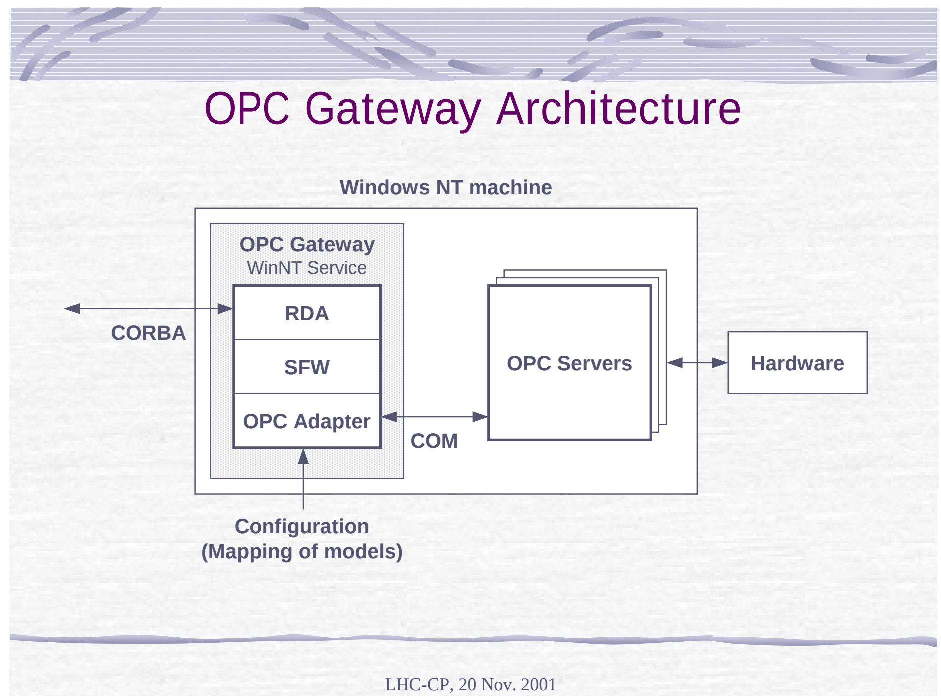
CMW Architecture




CMW Architecture
User written
Middleware
Existing or off-shelf
Java Control Programs
RAD programs (VB, Excel)
C Programs
Clients
Middleware Client API
Device/Property Model
Topic Model
Naming Service
Configu-ration Service
CMW Infrastructure
CORBA, RDA, JMS
Server Framework API (C++, Java)
Servers
Virtual Devices (Java, C++)
Power Converter
SL-Equip Server
PS-GM Server
OPC Gateway
Physical Devices (RF, BT, BI, Powering)
LHC-CP, 20 Nov. 2001


CMW Architecture
User written
Middleware
Existing or off-shelf
Java Control Programs
RAD programs (VB, Excel)
C Programs
Clients
Middleware Client API
Device/Property Model
Topic Model
Naming Service
Configu-ration Service
CMW Infrastructure
CORBA, RDA, JMS
Server Framework API (C++, Java)
Servers
Virtual Devices (Java, C++)
Power Converter
SL-Equip Server
PS-GM Server
OPC Gateway
Physical Devices (RF, BT, BI, Powering)
LHC-CP, 20 Nov. 2001

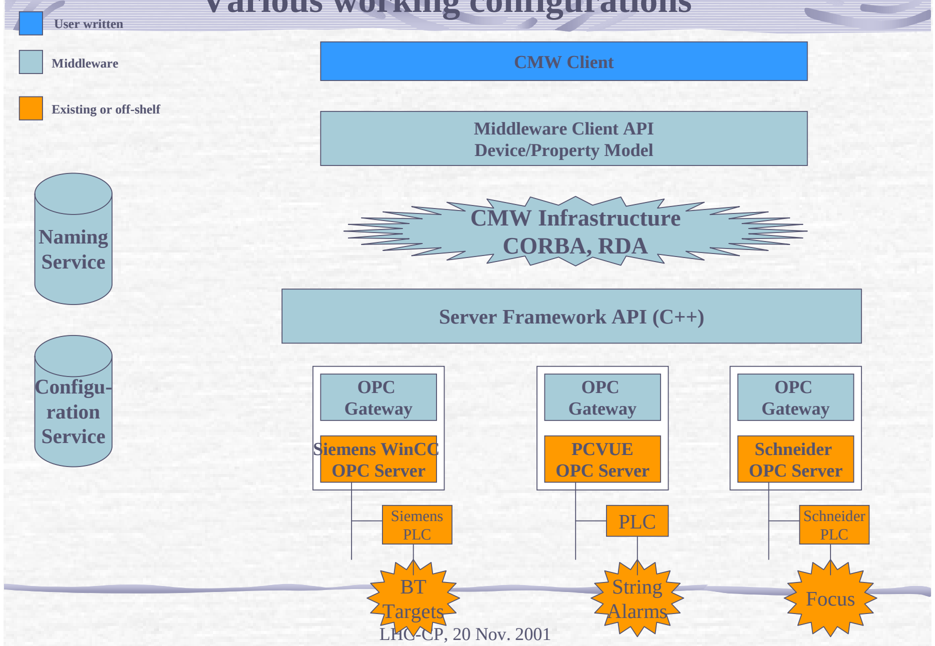

OPC in CMW Architecture
User written
Middleware
Existing or off-shelf
Java Control Programs
Middleware Client API
CORBA client adapter in Java
Naming Service
Get/Set CORBA
CORBA server adapter in C++
Middleware Server Framework (C++)
OPC Device Server in C ++
Windows NT
Visual C++
Configu-ration Service
OPC server(s)
LHC-CP, 20 Nov. 2001


OPC Gateway Architecture
Windows NT machine
OPC Gateway
WinNT Service
RDA
SFW
OPC Adapter
CORBA
COM
OPC Servers
Hardware
Configuration
(Mapping of models)
LHC-CP, 20 Nov. 2001

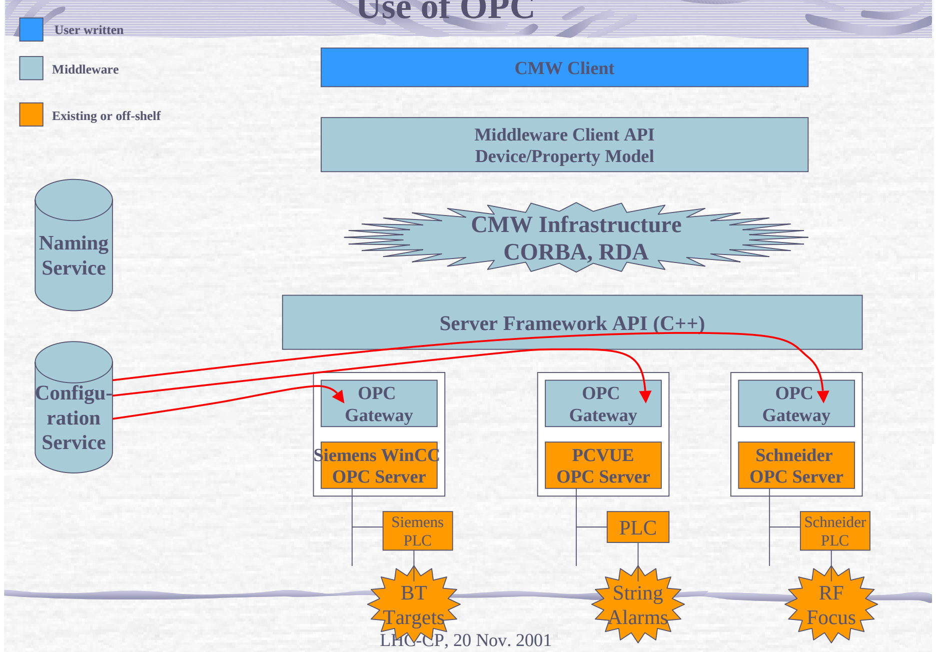
3 different types of OPC servers have been tried

- **Siemens WinCC** - SPS BT Targets
- **PCVUE** – First a test installation, recently String2 alarms
- **Schneider** – RF LHC Focus control

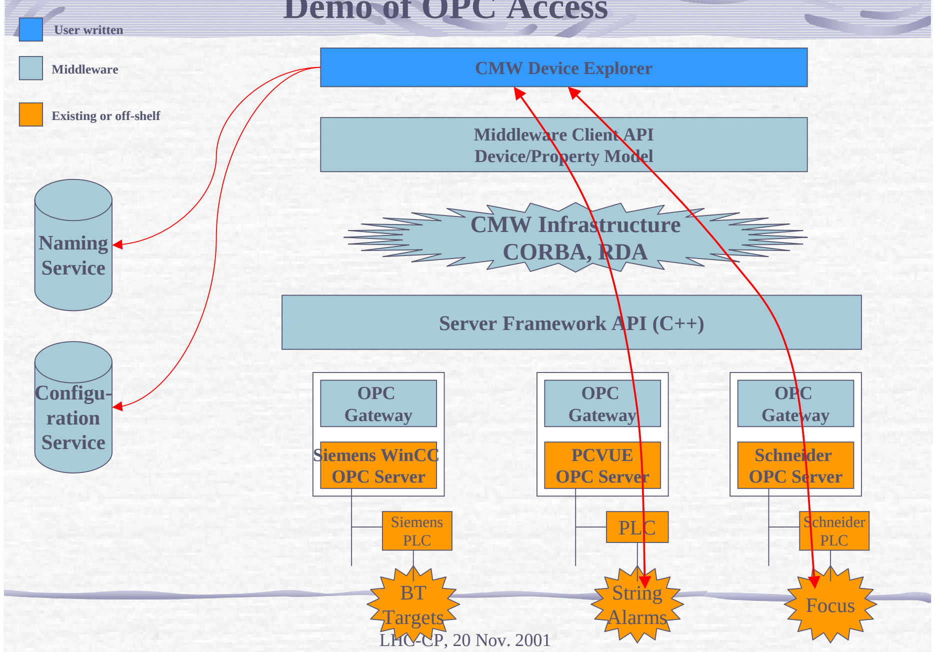
Various working configurations

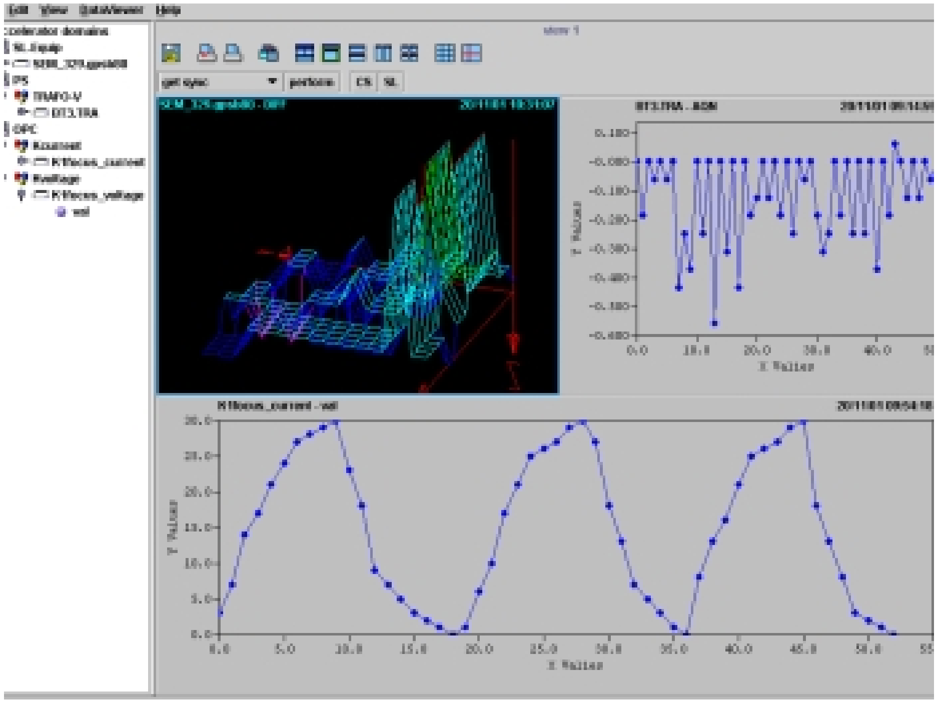


Use of OPC



Demo of OPC Access





Device/Property Model versus OPC Model

- Control system consists of ***named devices*** (position monitor, beam line)
- Devices are composed of ***properties*** (position, current)
- Operations on properties ***set, get, subscribe, unsubscribe***
- Devices organized into ***device classes***

- **OPC Server** has a number of **OPC ID's** (naming space)
- Clients access resources by terms of **OPC Items** organized into **OPC Groups**
- OPC Group defines characteristics of access (frequency, deadband)

OPC Gateway Configuration in Database

- Mapping between Device/Property and OPC item ID's
  - K1focus_voltage + val -> h112ctrl2!K1focus_voltage_val
  - CRYO.ALARMS.PT9605 + Status -> CRYO.ALARMS.ALWAYS.PT9605.STATUS
- Definition of Classes and Properties
  - Device K1focus_voltage belongs to class Kvoltage
  - Class Kvoltage has properties val, allow, alhi, etc.
  - Property val is of type long int, access is read only
- Definition of Items and Groups
  - Item h112ctrl2!K1focus_voltage_val will be in a group reporting every 1000ms, deadband is 0 (report every change)