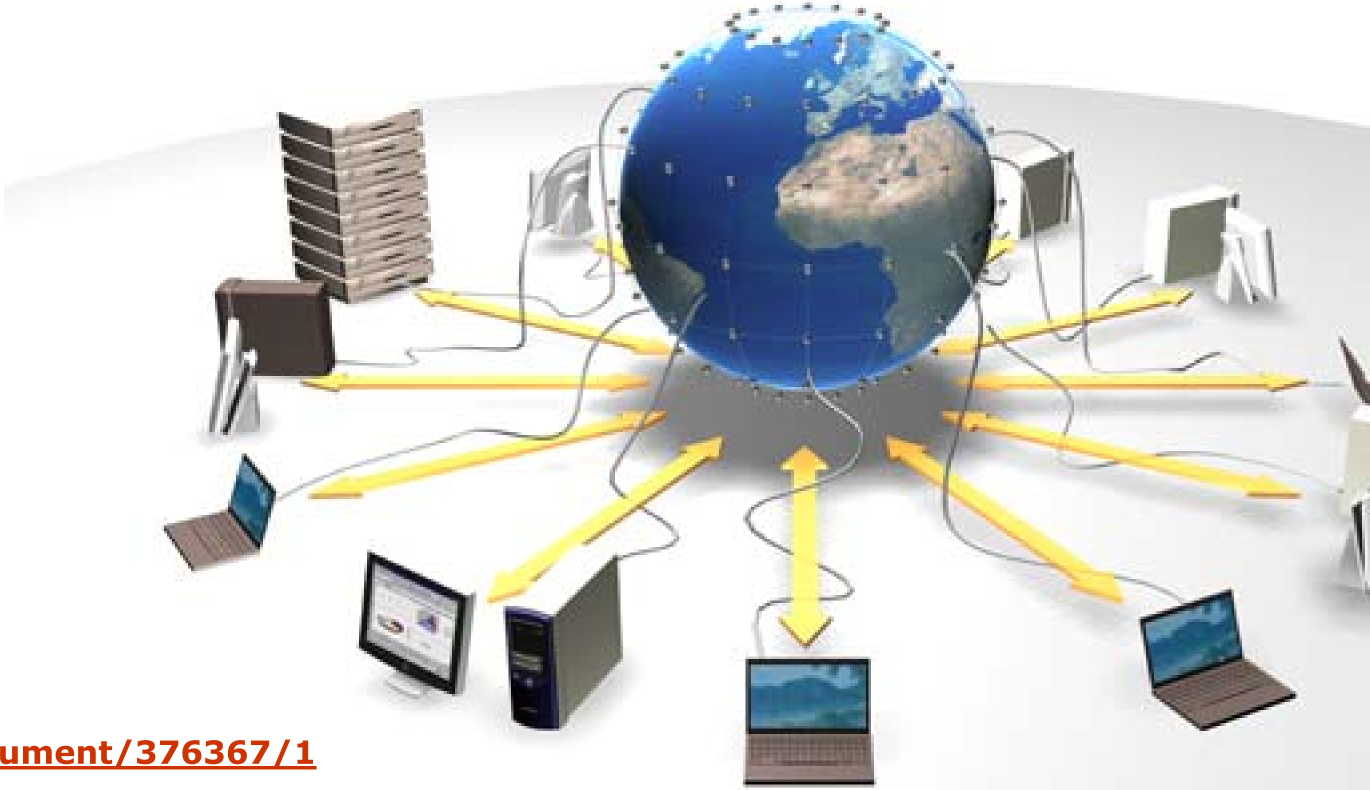
owards automation of computing fabrics using ools from the fabric management workpackage f the EU DataGrid project

aite Barroso Lopez
WP4)

aite.Barroso.Lopez@cern.ch



tps://edms.cern.ch/document/376367/1

Talk Outline

- Introduction to EU DataGrid workpackage 4
- Automated management of large clusters
- Components design and development status
- Summary and outlook

Authors

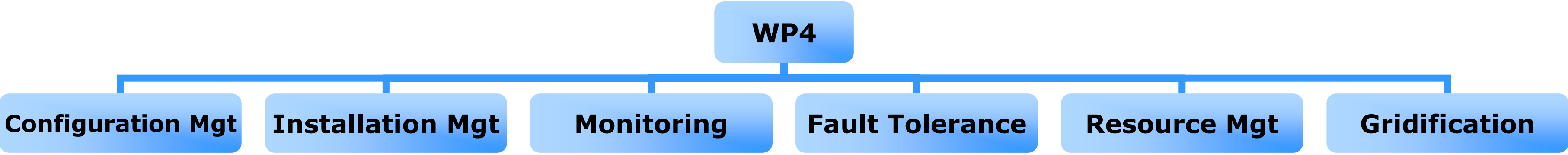
Olof Bärring, Maite Barroso Lopez, German Cancio, Sylvain Chapeland, Lionel Cons,
Piotr Poznański, Philippe Defert, Jan Iven, Thorsten Kleinwort, Bernd Panzer-Steinde
Jaroslaw Polok, Catherine Rafflin, Alan Silverman, Tim Smith, Jan Van Eldik - CERN
Massimo Biasotto, Andrea Chierici, Luca Dellagnello, Gaetano Maron, Michele
Michelotto, Cristine Aiftimiei, Marco Serra, Enrico Ferro – INFN
Thomas Röblitz, Florian Schintke – ZIB
Lord Hess, Volker Lindenstruth, Frank Pister, Timm Morten Steinbeck – EVG UNI HE
David Groep, Martijn Steenbakkers – NIKHEF/FOM
Paul Anderson, Tim Colles, Alexander Holt, Alastair Scobie, Michael George - PPARC

WP4 objective and partners

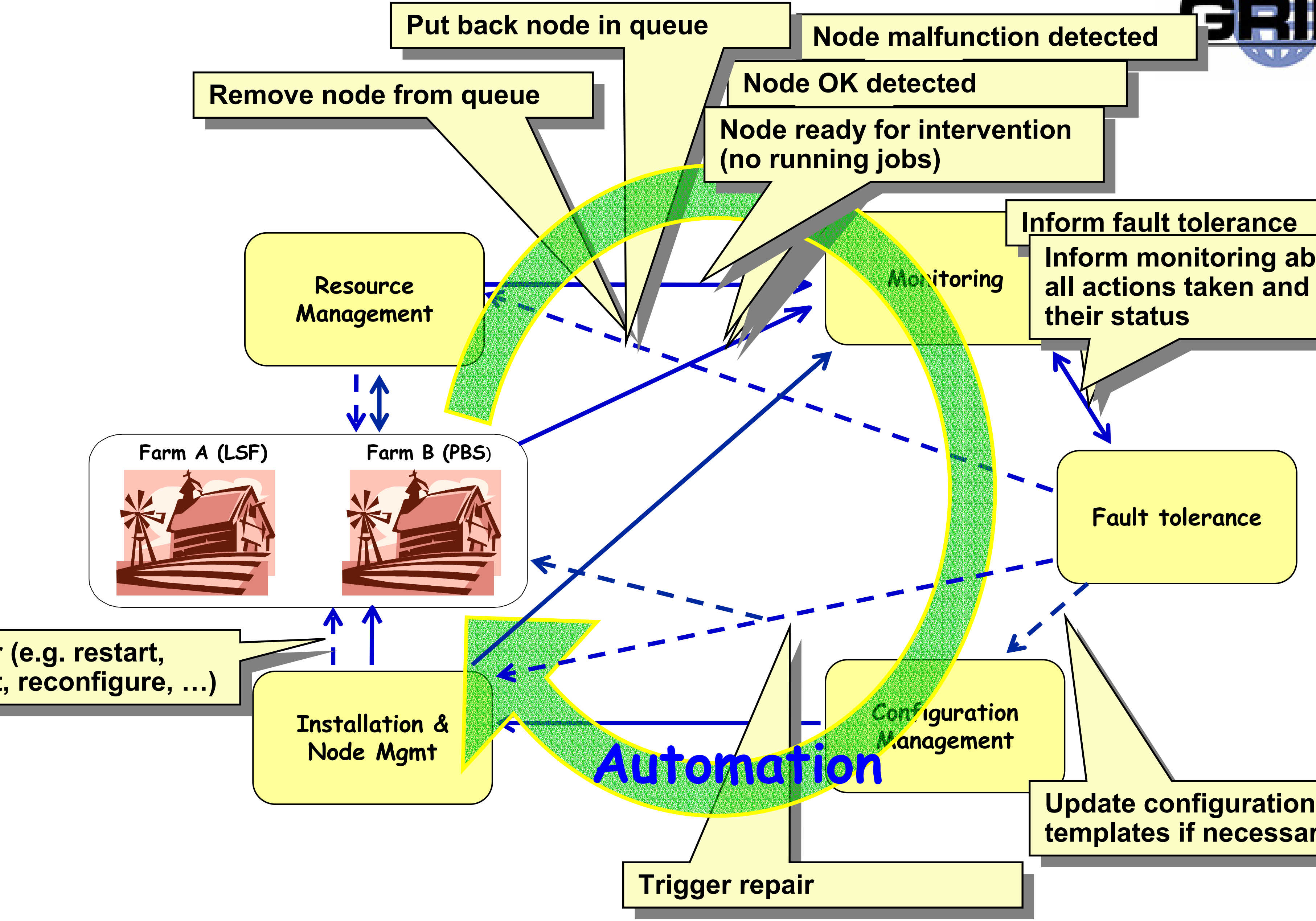
‘To deliver a computing fabric comprised of all the necessary tools to manage a center providing grid services on clusters of thousands of nodes.”

- User job management (Grid and local)
- Automated management of large clusters

- 6 partners: CERN, NIKHEF, ZIB, KIP, PPARC, INFN.
- ~14 FTEs (6 funded by the EU).
- The development work divided into 6 subtasks:



Automated management of large clusters



Put back node in queue

Node malfunction detected

Node OK detected

Remove node from queue

Node ready for intervention (no running jobs)

Inform fault tolerance

Inform monitoring ab all actions taken and their status

Resource Management

Monitoring

Farm A (LSF)

Farm B (PBS)

Fault tolerance

r (e.g. restart, t, reconfigure, …)

Installation & Node Mgmt

Configuration Management

Automation

Update configuration templates if necessar

Trigger repair

Monitoring subsystem: design

Monitoring Sensor Agent

- Calls plug-in sensors to sample configured metrics
- Stores all collected data in a local disk buffer
- Sends the collected data to the global repository

Transport

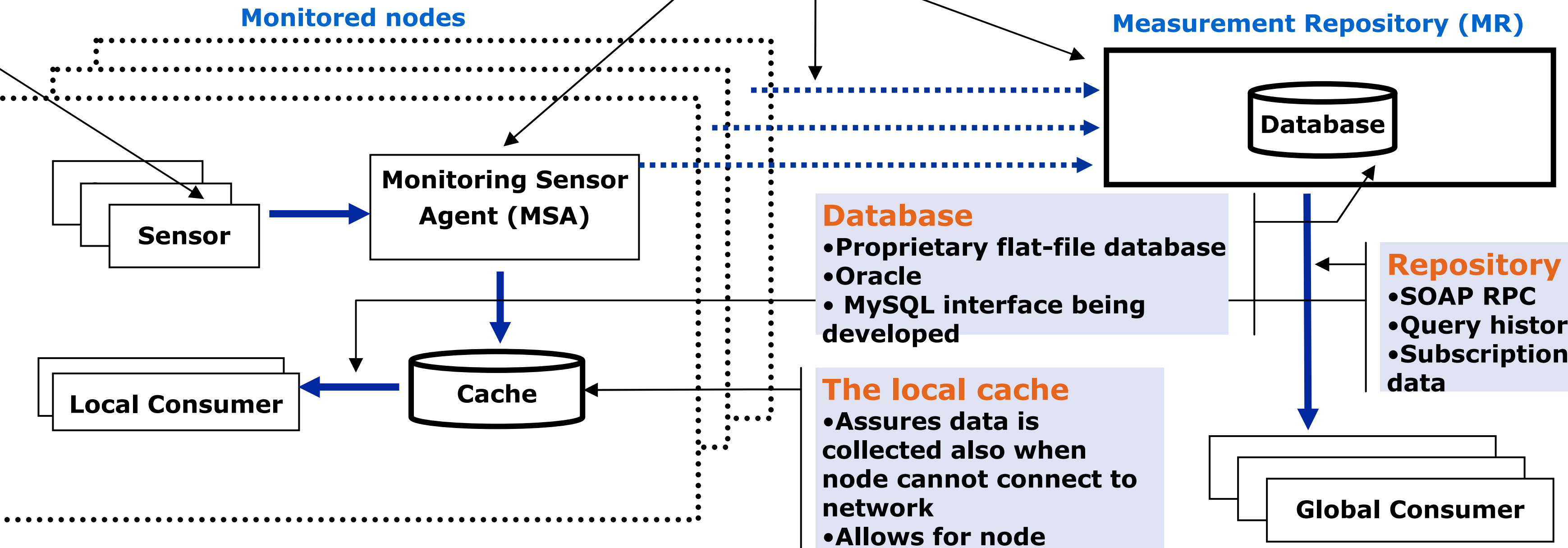
- Transport is pluggable.
- Two proprietary protocols over UDP and TCP are currently supported where only the latter can guarantee the delivery

Plug-in sensors

- Programs/scripts that implements a simple sensor-agent ASCII text protocol
- A C++ interface class is provided on top of the text protocol to facilitate implementation of new sensors

Measurement Repository

- The data is stored in a database
- A memory cache guarantees fast access to most recent data, which is normally what is used for fault tolerance correlations.



Database

- Proprietary flat-file database
- Oracle
- MySQL interface being developed

Repository API

- SOAP RPC
- Query history dat
- Subscription to ne data

The local cache

- Assures data is collected also when node cannot connect to network
- Allows for node autonomy for local repairs

Monitoring subsystem: status

- Local nodes:
  - Monitoring Sensor Agent (MSA) and UDP based proprietary protocol ar ready and used on CERN production clusters since more than a year.
  - The TCP based proprietary protocol exists as prototype. Extensive testing needed to be ready for production use.
- Central services
  - Repository server exists with both flatfiles and Oracle database. Suppo for MySQL is planned for this summer.
  - Alarm display: still in early prototype phase.
- Repository API for local and global consumers:
  - C library implementation of API (same for local and global consumers)
  - Bindings for other languages can probably be generated directly from the WSDL

Fault tolerance subsystem: design

Feedback to monitoring

- Actuator agent is reporting the result of the correlation and repair action (if any) to the monitoring system
- This feedback is important for tracing of all exceptions and repairs

Decision unit

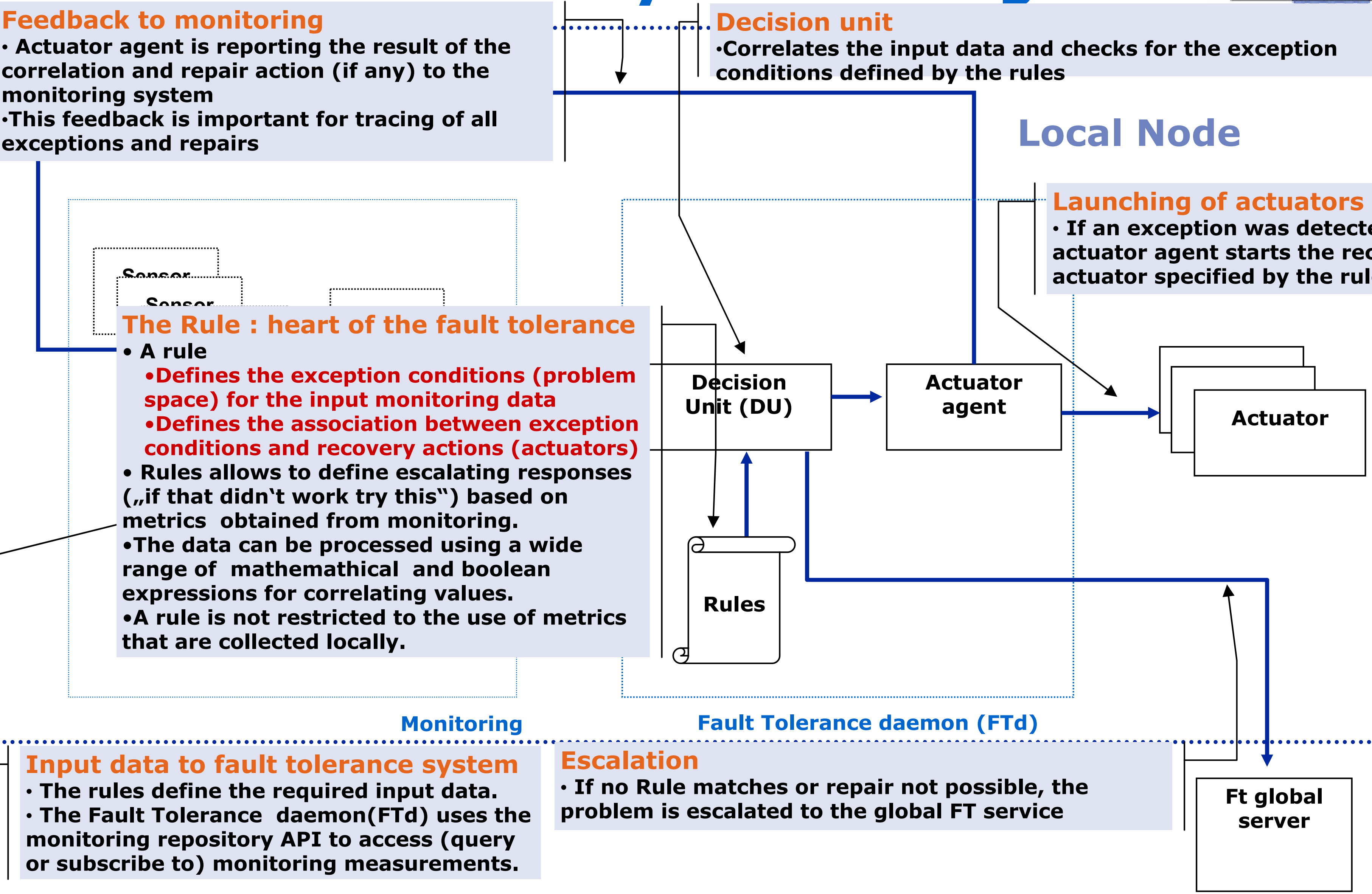
- Correlates the input data and checks for the exception conditions defined by the rules

Launching of actuators

- If an exception was detecte
actuator agent starts the rec
actuator specified by the rul

The Rule : heart of the fault tolerance

- A rule
  - Defines the exception conditions (problem space) for the input monitoring data
  - Defines the association between exception conditions and recovery actions (actuators)
- Rules allows to define escalating responses („if that didn't work try this") based on metrics obtained from monitoring.
- The data can be processed using a wide range of mathematical and boolean expressions for correlating values.
- A rule is not restricted to the use of metrics that are collected locally.



Input data to fault tolerance system

- The rules define the required input data.
- The Fault Tolerance daemon(FTd) uses the monitoring repository API to access (query or subscribe to) monitoring measurements.

Escalation

- If no Rule matches or repair not possible, the problem is escalated to the global FT service

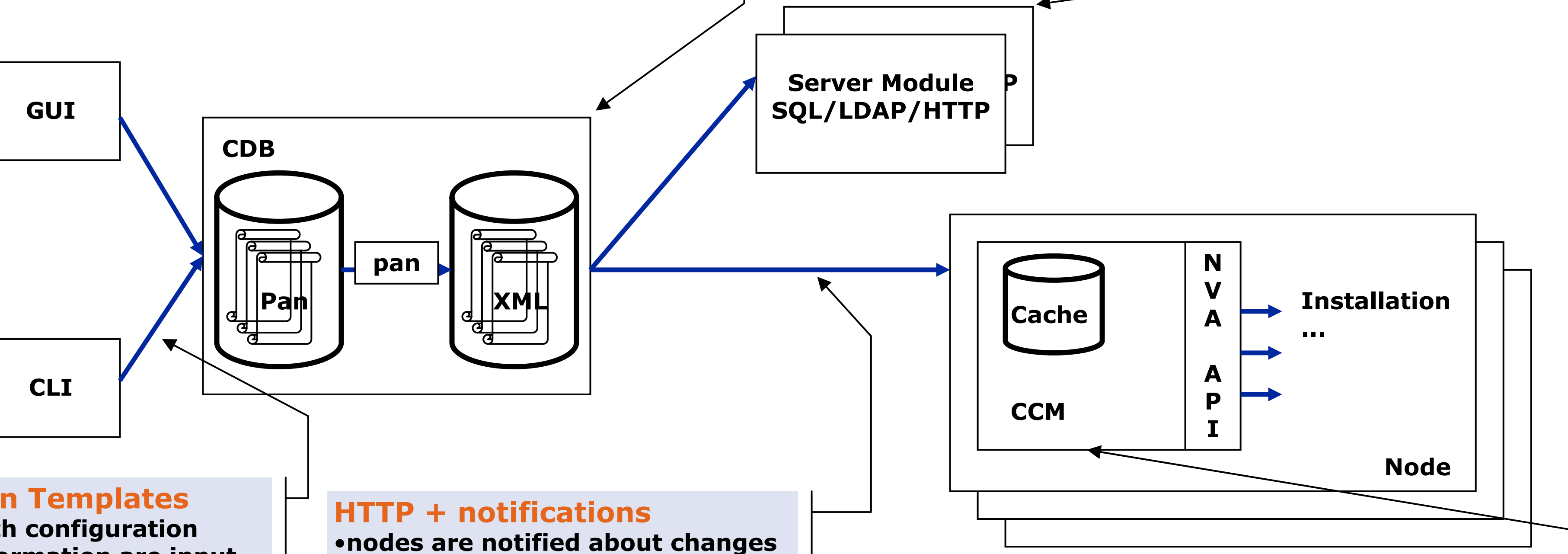
Fault tolerance subsystem: status

- Not yet ready for production deployment
- Prototype was demonstrated working together with the fabric monitoring system at EU review in February 2003
  - Web-based rule editor
  - Central Rule repository (MySQL)
  - Local FTd (fault tolerance daemon) that
    - Automatically subscribes to monitoring metrics specified by the rules
    - Launches the associated actuators when the correlation evaluates to an exception
    - Reports back to the monitoring system the recovery actions taken and their status
  - Global correlations not yet supported

Configuration Management subsystem: design

Configuration Data Base (CDB)
Configuration Information store. The information is updated in transactions, it is validated and versioned. Pan Templates are compiled into XML profiles

Server Modules
Provide different access patterns to Configuration Information



n Templates
h configuration ormation are input o CDB via GUI & CLI

HTTP + notifications
- **nodes are notified about changes of their configuration**
- **nodes fetch the XML profiles via HTTP**

Configuration Information is stored in the local cache. It is accessed via NVA-API

Configuration Management subsystem: status

- System in implemented (except for CLI and Server Modules), most of the components in 1.0 production version,
- Pilot deployment of the complete system at CERN production clusters, using the “panguin” GUI (screenshot next slide)

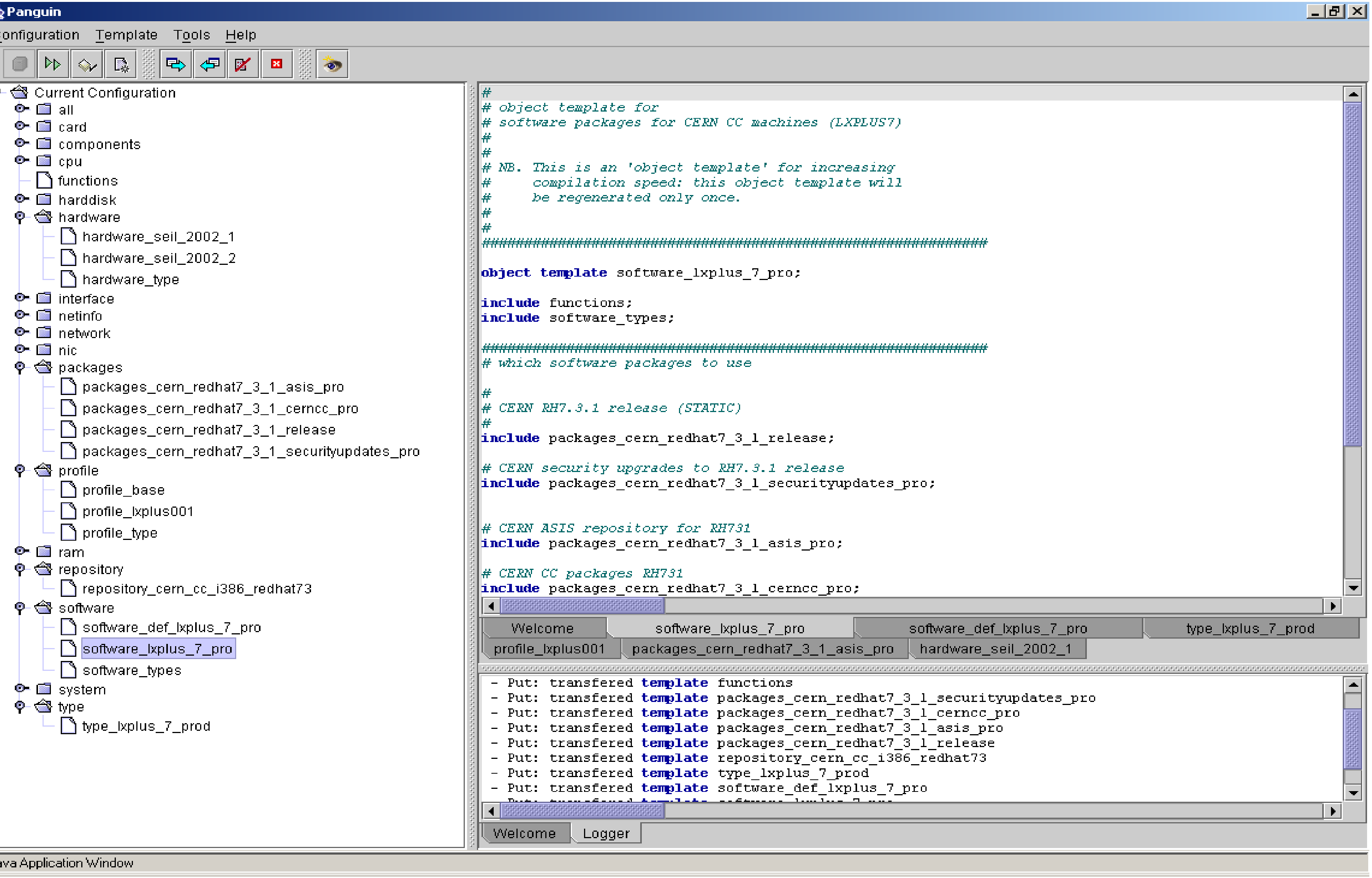
In parallel:

- System being consolidated,
- Issues of scalability and security being studied and addressed,
- Server Modules under development (SQL).

More information:

http://cern.ch/hep-proj-grid-config/

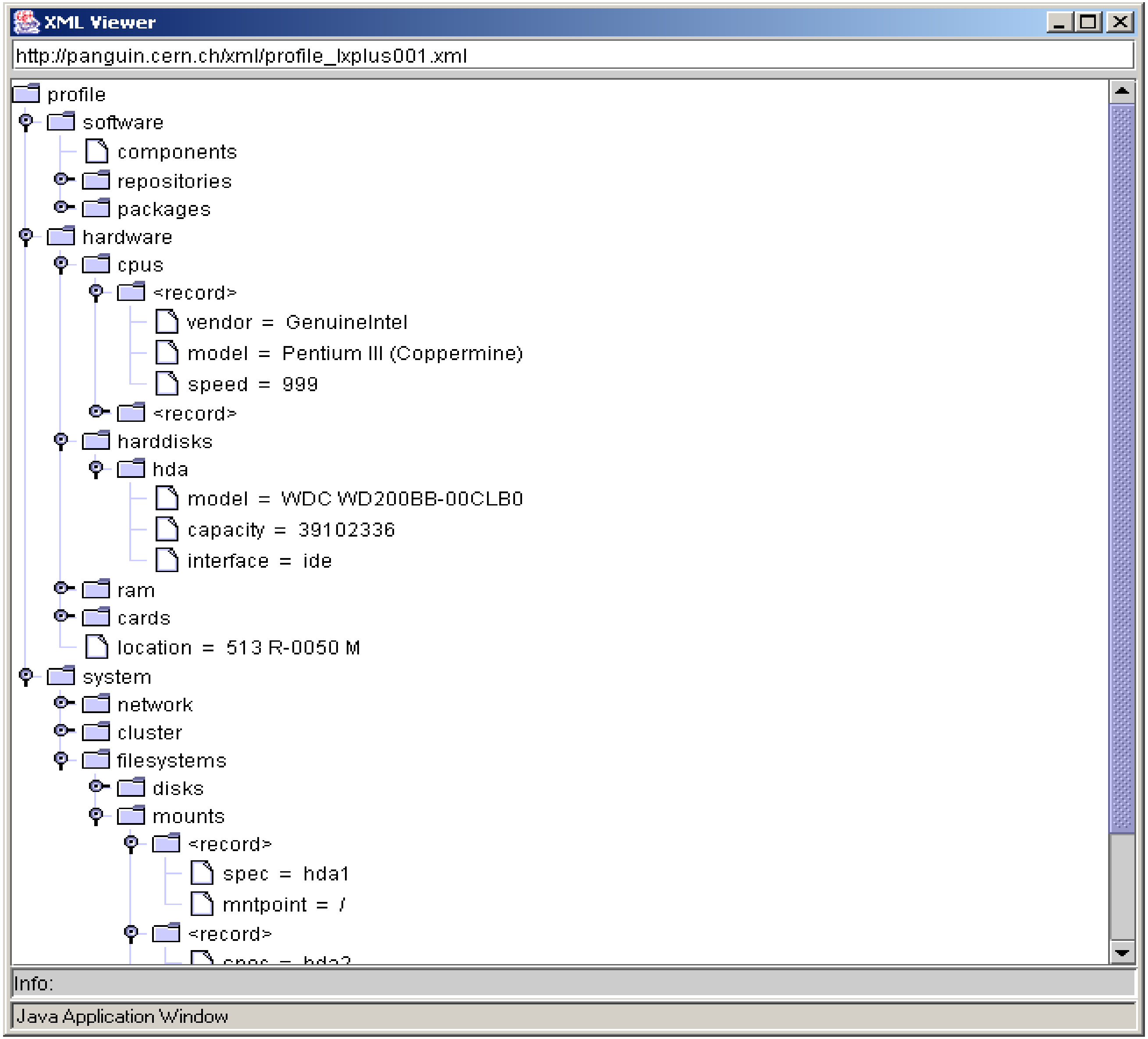
panguin GUI for managing/editing PAN templates (Courtesy: Martin Murth, CERN-IT/FIO)



```
#
# object template for
# software packages for CERN CC machines (LXPLUS7)
#
#
# NB. This is an 'object template' for increasing
#     compilation speed: this object template will
#     be regenerated only once.
#
#
############################################################

object template software_lxplus_7_pro;

include functions;
include software_types;

############################################################
# which software packages to use

#
# CERN RH7.3.1 release (STATIC)
#
include packages_cern_redhat7_3_1_release;

# CERN security upgrades to RH7.3.1 release
include packages_cern_redhat7_3_1_securityupdates_pro;


# CERN ASIS repository for RH731
include packages_cern_redhat7_3_1_asis_pro;

# CERN CC packages RH731
include packages_cern_redhat7_3_1_cerncc_pro;
```

```
- Put: transfered template functions
- Put: transfered template packages_cern_redhat7_3_1_securityupdates_pro
- Put: transfered template packages_cern_redhat7_3_1_cerncc_pro
- Put: transfered template packages_cern_redhat7_3_1_asis_pro
- Put: transfered template packages_cern_redhat7_3_1_release
- Put: transfered template repository_cern_cc_i386_redhat73
- Put: transfered template type_lxplus_7_prod
- Put: transfered template software_def_lxplus_7_pro
```

XML profile generated by PAN for a typical node (lxplus001)



http://panguin.cern.ch/xml/profile_lxplus001.xml

```
profile
  software
    components
    repositories
    packages
  hardware
    cpus
      <record>
        vendor = GenuineIntel
        model = Pentium III (Coppermine)
        speed = 999
      <record>
    harddisks
      hda
        model = WDC WD200BB-00CLB0
        capacity = 39102336
        interface = ide
    ram
    cards
    location = 513 R-0050 M
  system
    network
    cluster
    filesystems
      disks
      mounts
        <record>
          spec = hda1
          mntpoint = /
        <record>
          spec = hda2
```

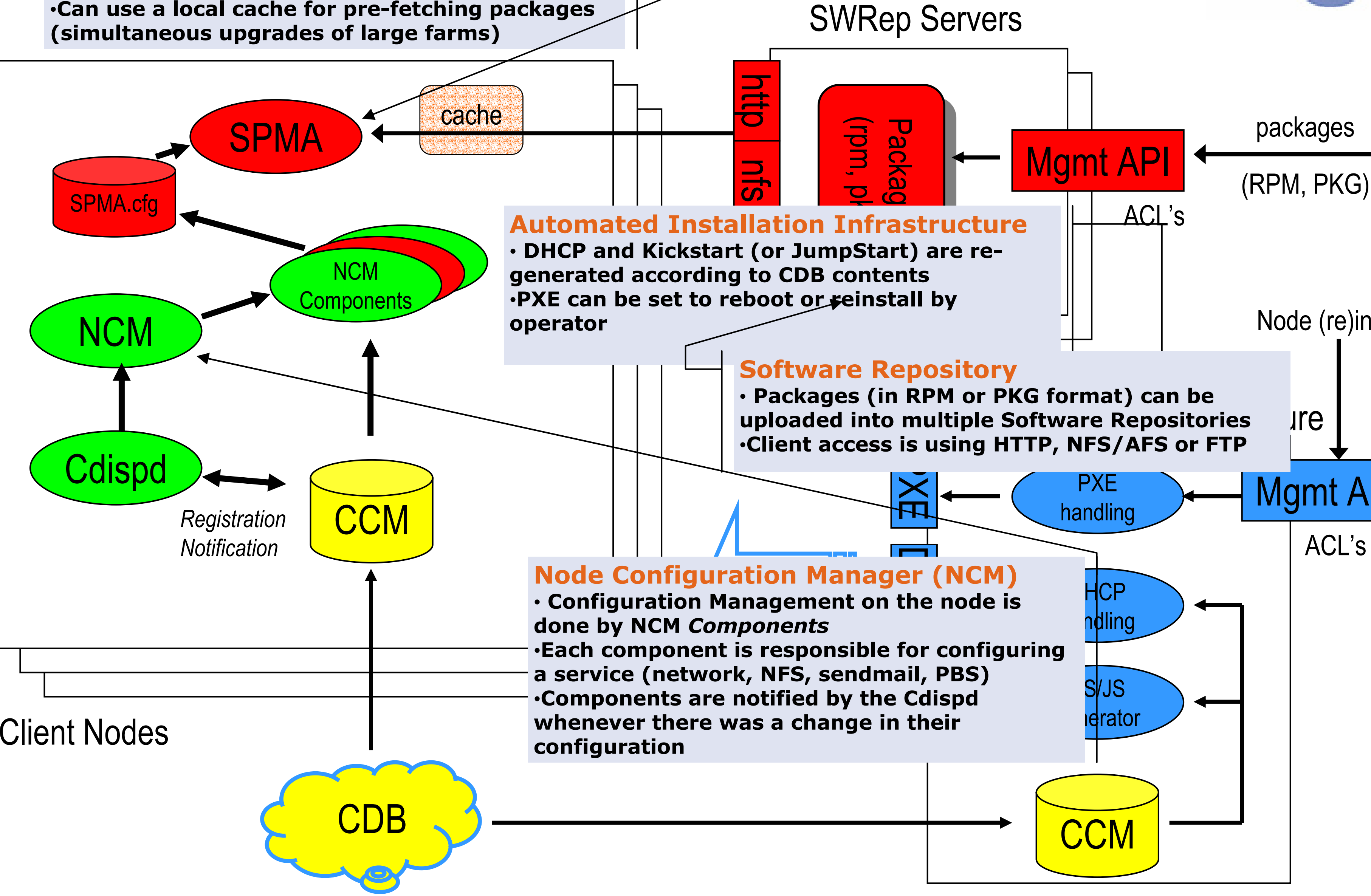
Info:

Java Application Window

In ment: design

Software Package Mgmt Agent (SPMA)

- SPMA manages the installed packages
- Runs on Linux (RPM) or Solaris (PKG)
- SPMA configuration done via an NCM component
- Can use a local cache for pre-fetching packages (simultaneous upgrades of large farms)



Automated Installation Infrastructure

- DHCP and Kickstart (or JumpStart) are re-generated according to CDB contents
- PXE can be set to reboot or reinstall by operator

Software Repository

- Packages (in RPM or PKG format) can be uploaded into multiple Software Repositories
- Client access is using HTTP, NFS/AFS or FTP

Node Configuration Manager (NCM)

- Configuration Management on the node is done by NCM *Components*
- Each component is responsible for configuring a service (network, NFS, sendmail, PBS)
- Components are notified by the Cdispd whenever there was a change in their configuration

Installation subsystem: status

- Software Repository and SPMA
  - First pilot being deployed on CERN Computer Centre for the central CERN production (batch & interactive) services
- Node Configuration Manager (NCM)
  - Design/development phase
  - Implementation available in Q2 2003
- Automated Installation Insfrastructure (AII)
  - Design/development phase
  - Linux Implementation expected for Q2 2003

Summary & Future Work

- Experience and feedback with existing tools and prototypes helped to get requirements and early feedback from users

- First implementation now ready for all the subsystems

- Some of them already deployed at CERN and/or EDG testbed. The rest wil come during this year.

- Close collaboration with CERN service managers and LCG

- What is still missing:
  - General: Scalability, Security, GUIs
  - Integration between the different fabric subsystems to build a consistent set of fabric management tools
  - From prototype to production quality

Thanks to the EU and our national funding agencies for their suppor of this work