

Automation and Control

CERN OpenLab
25 January 2010



- **PVSS**
 - Openlab staff: D. RODRIGUES
 - Openlab fellow: M. BOGUSZ
 - CERN tech. sup.: M. GONZALES

- **Step7**
 - Openlab fellow: O. KHALID
 - CERN tech. sup.: R. BARILLERE

- **Security and control devices**
 - Openlab fellow: F. TILARO
 - CERN tech. sup.: B. COPY



- Topics
 - PVSS
 - Archiving
 - Development Environment
 - Web Access
 - Step 7
 - Deployment / Security
 - PVSS Security
 - Security and Control Devices

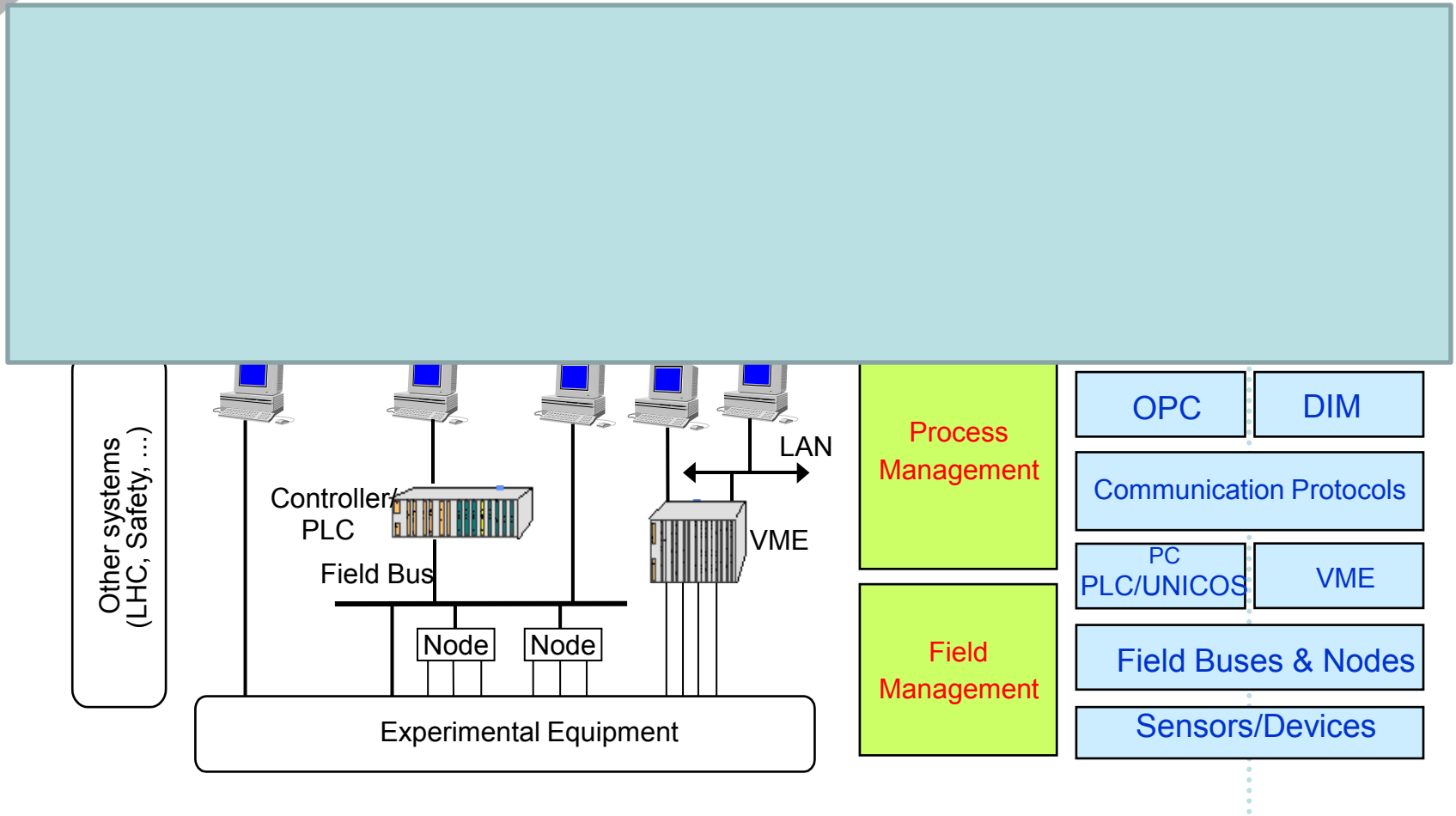


PVSS

- What is SCADA
 - Supervisory control and data acquisition software
 - Supervisory layer on top of the control system

- What is PVSS
 - Commercial SCADA product from ETM, Austria
 - Widely adopted across CERN
 - Some features
 - Human-machine interface (process visualization)
 - Devices described as data points (DP), device properties as data point elements (DPe)
 - Control device access
 - Alarm handling (display, filtering, etc.)
 - Archiving and trending

Layer Structure Technologies



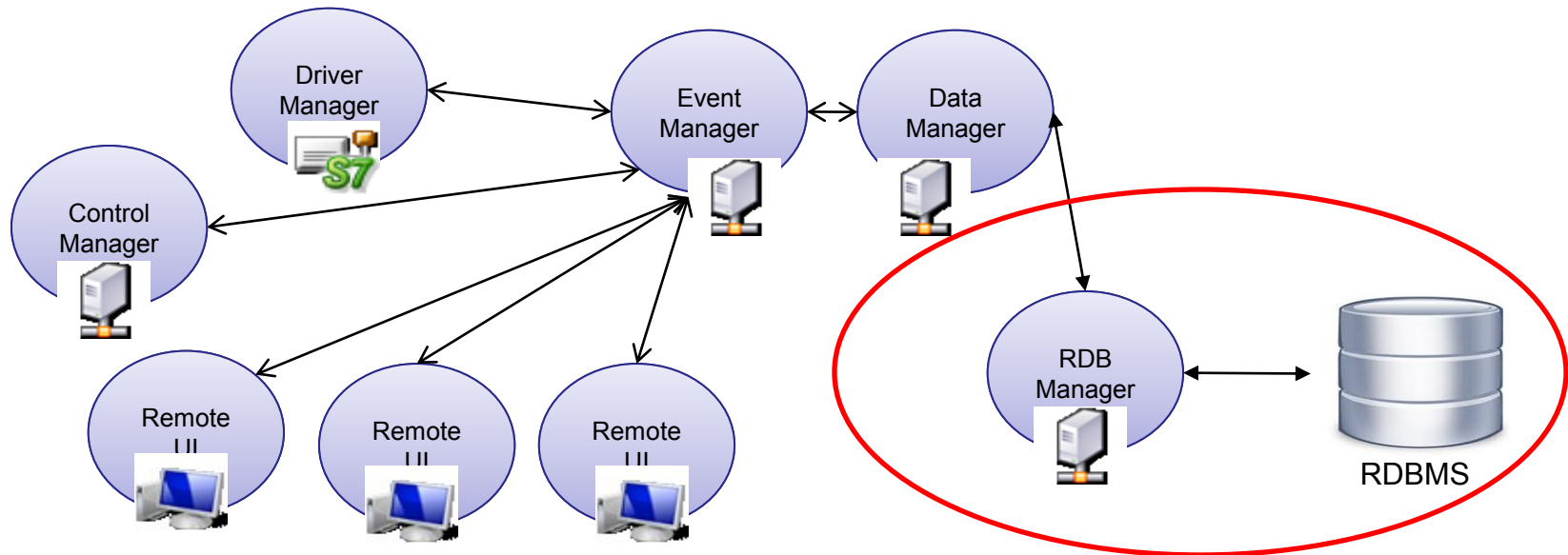
- Applications at CERN
 - Experiments' Detector Control Systems
 - LHC Cryogenics
 - Cooling & Ventilation
 - Etc.

- Present at CERN since 2000

- Long term fruitful collaboration between CERN and ETM ever since

- CERN has put years of effort in extending PVSS to it's custom needs (JCOP framework as an example)

- Oracle database archiver for the upcoming ETM SCADA system
 - A new project, started last year
 - Involves ETM (Siemens subsidiary)
- Scope of work



- Archiving in PVSS
 - Long term storage of the data on a physical media eg. relational database
- Why do we archive
 - Trend plots for monitored values
 - Analytics (post mortem analytics, etc.)
 - Physics (conditions) data analysis
- Data being archived by CERN
 - Event values (DPe values) and alarms
- Numbers
 - ~150 machines for an experiment DCS (forming a PVSS distributed system)
 - All 150 machines use a single DB (150 connected clients)
 - Current system is scalable (based on Oracle RAC) and has proven a very high performance well enough to cover experiments' needs (in the lab conditions 250 K value changes/s have been stored for a few hours – a fruit of collaboration between CERN and ETM)
 - order of millions of datapoint elements per system, however not all of them are archived



- Major effort devoted to a new version (PVSS 4) to come out in a few years
- Totally new architecture of logging/archiving services
 - Xml configurable generic archiving component
 - Dynamically loaded storage modules for various RDMS implementing a common interface
- Logging services architecture designed to handle not only SCADA but also other Siemens Products
- Architecture comes from ETM, CERN would have an influence on it
- The goal is to develop the Oracle archiving module that fulfills CERN's requirements (e.g. at least no degradation compared to current version)
- Storage modules for other relational database products are developed by ETM, we are responsible for the development of Oracle solution



Why ETM (Siemens) want us to develop the new Oracle archiver ?

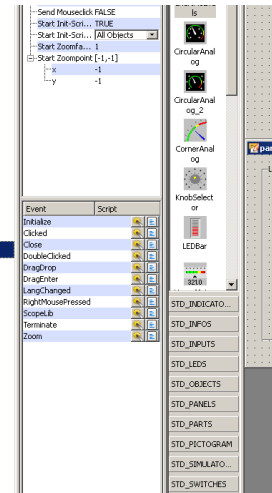
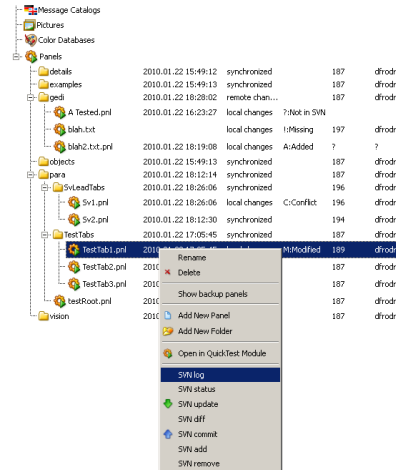
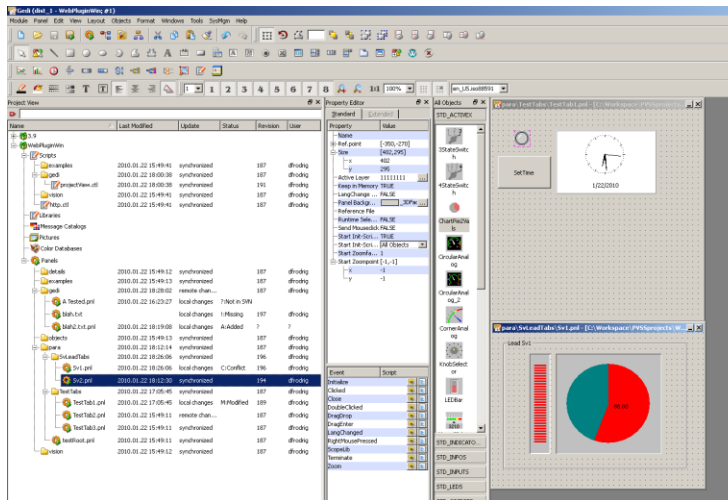
- Long term collaboration with CERN
 - CERN has been a major user of the Oracle archiver
 - CERN has helped to improve the Oracle archiving module to increase the performance
 - Many years of experience in archiving large amount of fast changing data – Oracle KNOW HOW
 - CERN runs several large distributed systems
- CERN will use the next version of PVSS
- Oracle archiving module could be used not only by the SCADA system, but given the flexible logging architecture, also in other Siemens products

- Close cooperation with ETM
- Study of the current archiving architecture (PVSS 3.X)
 - Data model
 - Changes performed by CERN to boost the performance
 - Potential area of improvements
- Review of the architectural documentation for the new (version 4) system
 - Frameworks
 - Components
 - New storage management architecture
- Implementation of the storage module for Oracle (OCCI/OCI) for further evaluation
 - Identification of areas for improvement – request/propose changes to ETM
- Testing and evaluation
 - Low level unit tests of the basic storage module functionalities (connection to the data source, performing statements)
 - Higher level tests using configured archiving component and PVSS database schema
- Performance evaluation

-  Integration in the ETM environment
 - Software frameworks
 - Test suits
-  Most of the basic functionalities implemented and unit tested
 - Statement handling
 - Bulk operations
 - Identified parts that would need improvements (to be discussed with ETM)
- Segmentation/Partitioning capabilities to be implemented
- Find solutions for identified issues with ETM
- Higher level tests with the PVSS database schema pending
- Performance tests

- **Problem scope:**
 - Developing a PVSS system creates a large quantity of code
 - panels, scripts, datapoint lists, configuration files.
 - An Integrated Development Environment exists for creating PVSS projects (GEDI)
 - Giving easy access to all key development tools
 - Limited aids for software management

- **Purpose of the topic:**
 - Integrate Versioning control of project files inside the GEDI



- Last Status
 - A prototype was made available
 - To CERN users and ETM
 - Using some elements of an initial version
 - Same libraries, using subversion command line.
 - Implementing key ideas for enhancement of GEDI
 - Enabling access to different sub projects
 - Diff tool for project files
 - Easy installation and start up

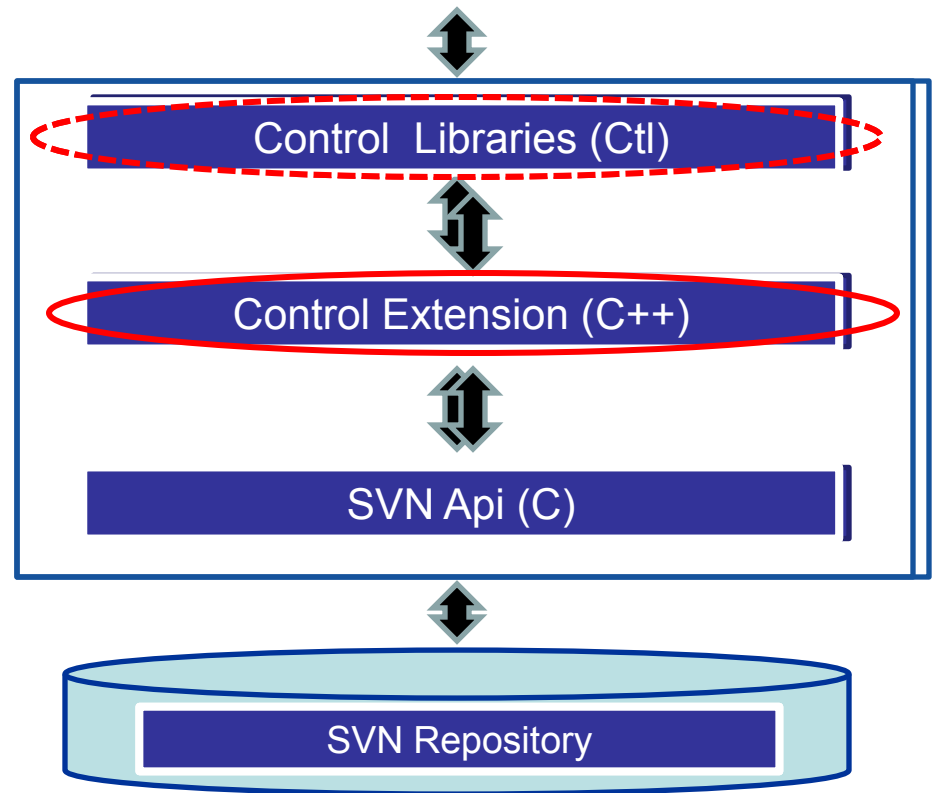
- Since then work focused on changes to the underlying libraries
 - To improve interaction with the Subversion repository.

- Most work done at the code level
 - No fancy updates to the GUI, sorry 😊
- Prototype used subversion command line
 - Has limitations
 - svn authentication contexts not under control
 - requiring use of pageant
 - no http based access to repository
 - Limited error handling and no server interaction
- Solution
 - A CtrlExt (Control Extension)
 - A PVSS feature, that allows adding custom C++ code to the PVSS native scripting language (Control)
 - Interfacing the subversion C API
 - More refined control in contacting the SVN repository
 - Possibility of managing the Subversion context within PVSS
 - Accessible from CTRL language

Project View

Name	Last Modified	Update	Status	Revision	User
3.9					
WebPluginWin					
Scripts					
examples	2010.01.22 15:49:41	synchronized		187	dfrodrig
gedi	2010.01.22 18:00:38	synchronized		187	dfrodrig
projectView.ctl	2010.01.22 18:00:38	synchronized		191	dfrodrig

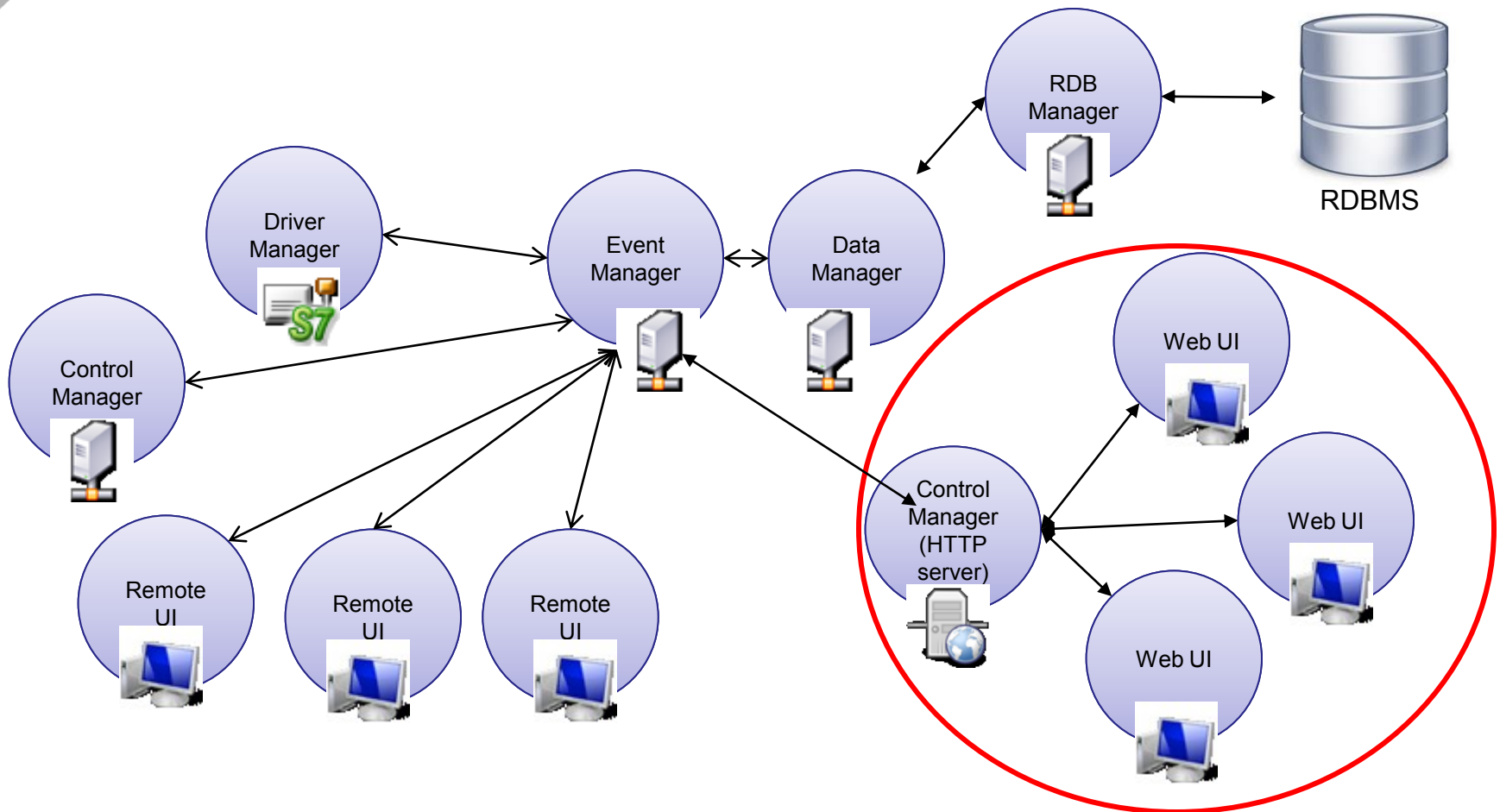
GEDI Project View



- Problem scope:
 - PVSS
 - Very flexible and highly distributed
 - Remote User Interfaces require a PVSS installation on each machine
 - ETM has made available a Web Client for version 3.9

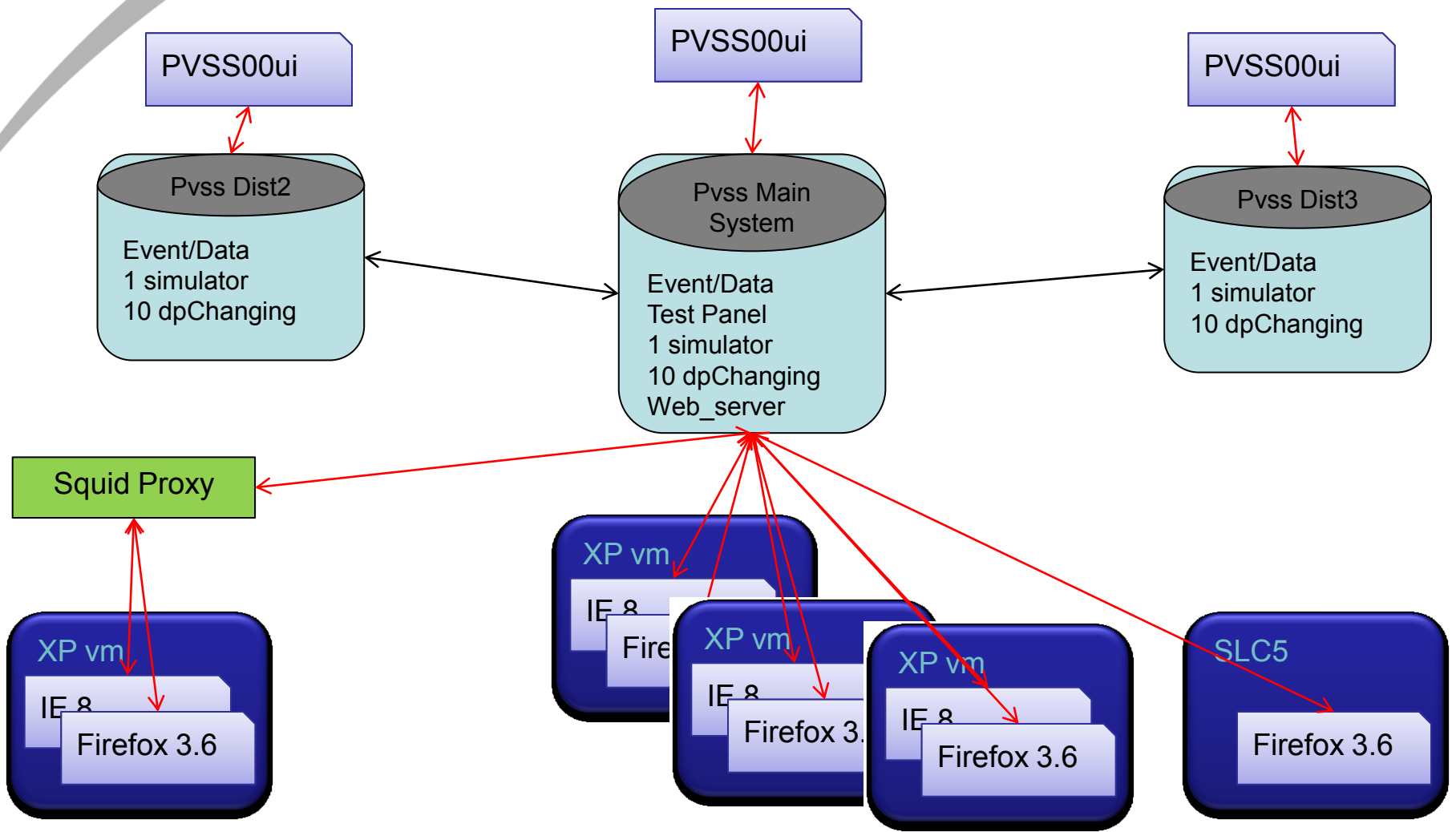
- Purpose of the topic
 - Test if the new Web Client is suitable for use at CERN
 - Does performance fit the general needs
 - Can it be used from outside the technical network
 - Test the performance in complex configurations and demanding loads

- Example of a PVSS System



- Last Status
 - Web Client initial release was tested
 - First conclusions reported
 - Installation; Basic functionality; Network Setup;
 - Performance (Stress tests)
 - A patch was made available by ETM addressing most of the issues, in particular:
 - Performance under Windows
 - Easier installation

- Since then work focused on retesting and extending tests to the patch.
 - Results were presented
 - Following the presentations, preparatory work for further testing was done.



- Upgraded scenario
 - 3 Distributed systems
 - Internal mapping of ~1 million dpValues :
 - 100 dpTypes
 - 1000 dpElements
 - 10 float/string values
 - Using Proxy Server (squid) from 2 clients

- Across 2 axis
 - Number of clients 1 – 14
 - 3 Remote UI's for cross checking
 - Up to 11 simultaneous Web Clients
 - Firefox Linux
 - Windows XP VM's with IE8 / Firefox 3.6
 - Different frequencies of update
 - 30 dpConnects (10 per system)
 - 3 Simulators, changing values every 1/2/5 ms

- **Results**
 - A very visible improvement on the performance of updates
 - Before, limited to ~200Hz frequency of update, now 4 times faster ~800Hz
 - Proxy test
 - Some issues under investigation, probably related to the proxy itself

- **Presentations**
 - JCOP, with representatives of ETM
 - A lot of questions raised, for further investigation
 - CPU Usage on each of the system managers and impact
 - Up to 100 Web Clients
 - Study firewall settings with CERN Security Team

- **Next Steps**
 - CtrlExtension for including CpuUsage on the tests (underway)
 - More focus on HTTPS and very large number of clients
 - Pocket Client testing (when considered stable by ETM)

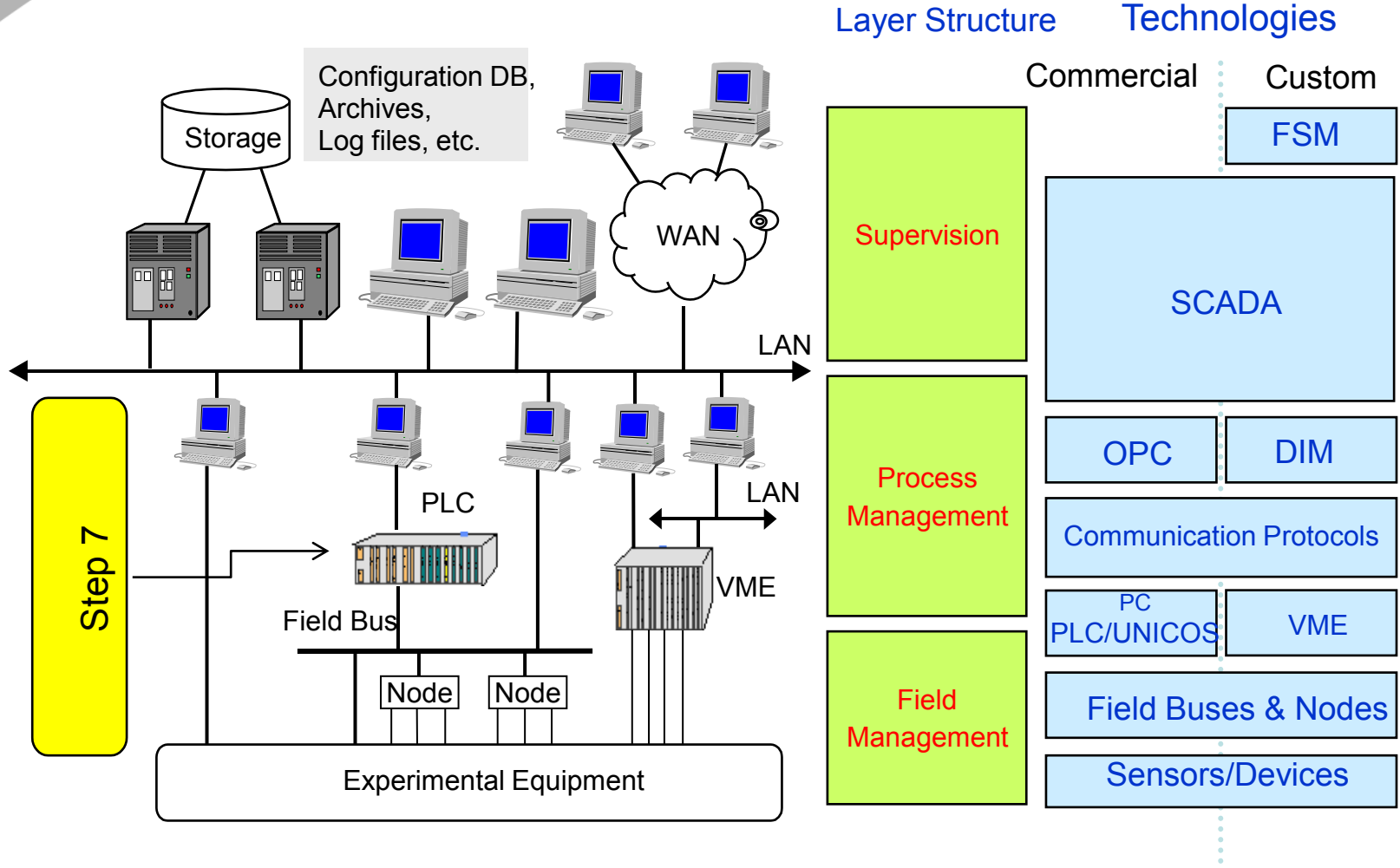
STEP7

PVSS Security

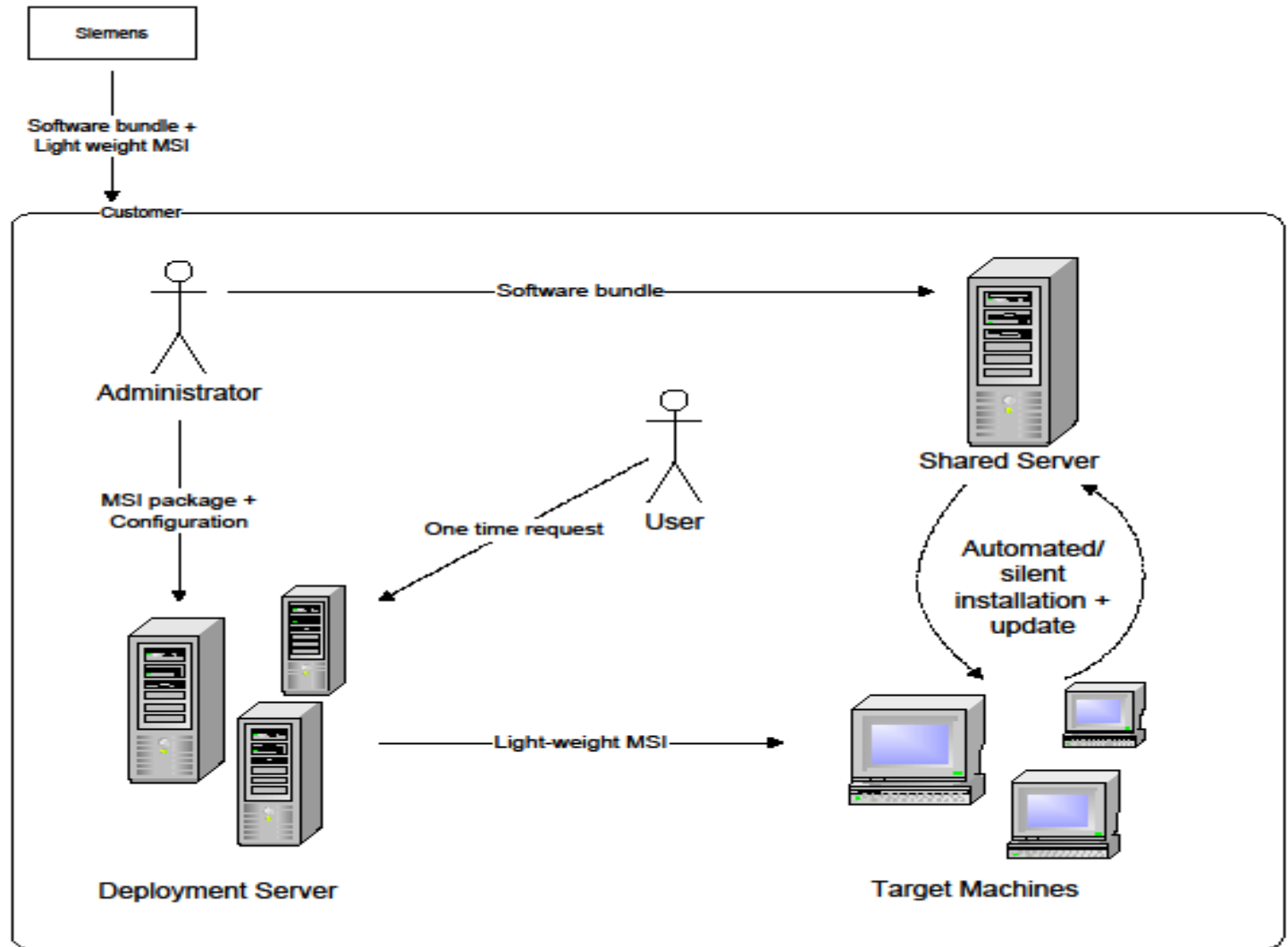
- Step 7 / Totally Integrated Automation:
 - Software development environment to develop software for PLC's that interfaces with the industrial equipment.

- Aim: To bring-in modern software engineering capabilities to Step7 product line:
 - Step7 Deployment
 - To automate the deploy Siemens software on engineering workstations; Scalability: from small (10's of machines) to large (100's of machines); Easy and flexible to deploy, fast refresh rate

Controls architecture



- **Status: Completed and Closed**
 - All milestones has been achieved and delivered. Verified and confirmed by Siemens.
- **Value for Siemens:**
 - Final strategy is implemented by Siemens in v12 of TIA.
 - TIA portal can now be deployed in automated fashion using 3rd party standard software inventory management software.
- **Approach:**
 - Three strategies validated through prototyping
 - Reported in detail in previous major review
 - Nutshell: either using chained MSI's or SIA engine
 - Meets short term, medium and long terms objectives and product development plans of Step7 software
 - Criteria: integration with Siemens existing software tools.



- Stuxnet worm
 - Detected in June 2010.
 - Attack method (0-day exploit against windows, fake certificates, rootkit, DLL replacement)
- Step7 Security
 - New topic was added to the project in Jul/Aug 2010
 - Test bench was setup
 - Market survey conducted – mostly source code based analysis
 - Binary code based analysis identified to complement existing source code based analysis
 - BitBlaze and Veracode selected as test candidates
- Status: Closed
 - Prior to any further testing/prototyping; Siemens decided to do this in-house.

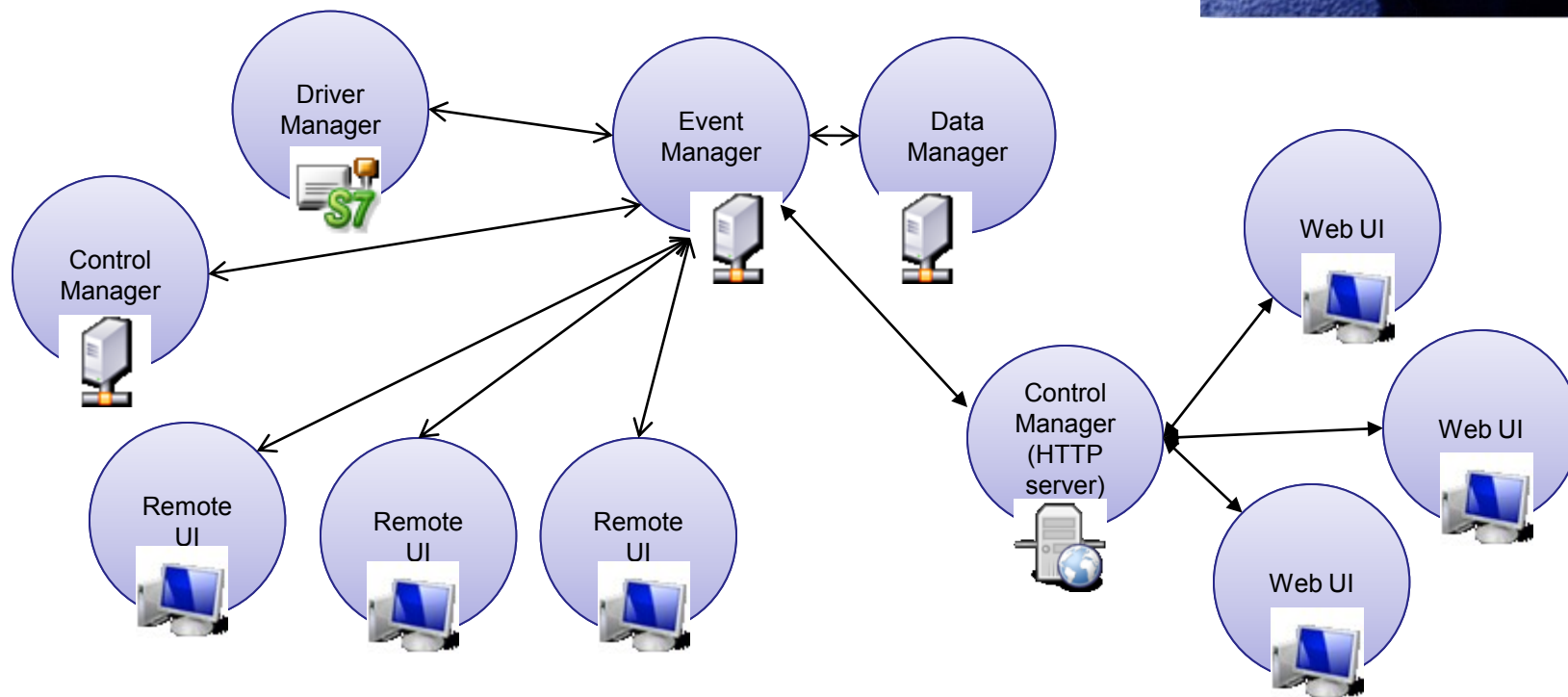




PVSS Security – extending PLC Security

■ PVSS

- Distributed architecture that uses TCP messages for synchronization
- Potentially vulnerable for network based attacks



- Started end of 2010
 - PVSS/SCADA systems are critical for LHC operations
 - Any vulnerability could potentially have serious consequences
 - In case an external agent manages to corrupt the status of the messages passed between various components of PVSS software
- Extending PLC Security:
 - Building on the experience of PLC security
 - i.e. tools, framework, testing approaches to penetrate PVSS TCP message communications
 - Initially using black box/external penetration to corrupt PVSS TCP messages
- Status: on-going work
 - Virtualization based test bench is deployed
 - PVSS software components distributed on a local network, and an attacker VM uses various
 - Initial tests are conducted and will be expanded in coming weeks.

- **Step7 Deployment**
 - SIA engine strategy implemented in v12; verified the strategy; Delivered. Topic closed.
- **Step7 Security**
 - After initial work, this topic has been closed and internalized by Siemens.
- **PVSS Security**
 - Applying to PVSS the “SCADA security guidelines” document produced by PLC security team, their tools, standards and testing approaches
 - On going work for the near future

Thank you for listening
QUESTIONS!

SECURITY AND CONTROL DEVICES

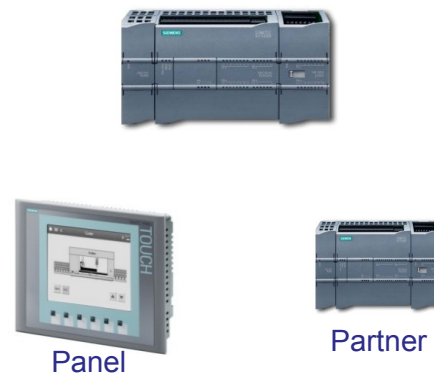


- **Objective**
 - Improve the Process Control System (PCS) security level
- **Importance of security**
 - In 2010 more than 180 security incidents were reported to the US Congress
 - Gas explosions, Deepwater Horizon, ...
 - Vulnerabilities in VxWorks and STUXNET
 - recovery from attacks is expensive! (loss of reputation, very costly time to market delays for vendors)
- **Strategy**
 - Design of a test-bench to discover and classify vulnerabilities
 - Determine key cyber security aspects relevant to CERN

System Testing



Target



System Monitoring



Reporting System



How to identify PLC failures?

- Loss of vision or control, process reaction time delays, loss of communication ... Each needs specific monitoring tools and technologies

Need for a generic interface between these monitoring tools and our test bench :

- Implementation of an Agent to react in case of failures and inform the test bench
- Implementation of a Watcher to save status and resume the testing process at the point of last failure

Delivery to Siemens:

- A Linux package to install and keep updated the Peach framework extensions
- Live-meeting with Siemens with a short demonstration on how to use the extended fuzzing framework

Siemens Bany PNIO as a part of the network monitoring

- Bany PNIO:
 - Based on FPGA (Field Programmable Gate Array) technology
 - Allows “real-time” traffic analysis and simulation for system testing
- We developed some Bany PNIO scripts to:
 - Analyze the packets coming in/out from the PLC target
 - Generate traffic at specific triggers
 - Replay of “malicious” packets sequences
- Limitations:
 - ✗ Hubbed network required
 - ✗ Bany PNIO fails to generate some Malformed Packets

Monitoring process and internal statuses

The I/O monitoring system

- Use of another Siemens PLC to:
 - Analyze the “target” PLC’s process signals
 - Detect any delays or anomalies during the testing phase
- ✘ The analysis is affected by synchronization issues between the two PLCs
- Investigation on the use of Digital Acquisition Cards (DAC) which could solve these timing issues

PLC internal status monitoring:

- Siemens S7 Linux library:
 - As a replacement of the open-source libnodave library
 - Supported by Siemens, also for the new PLC models
 - So far no API to retrieve diagnostic information from the PLC under test
 - Siemens will deliver an extension for this purpose in the next version of its SOFTNET library

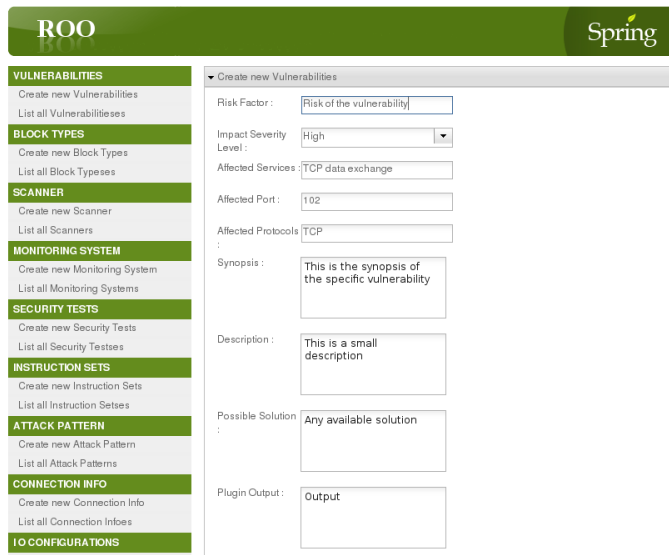
Achievements :

- Web user interface to insert and retrieve any previously stored vulnerabilities

Benefits :

- Follow-up on PLC Robustness, Debugging and Replay capabilities (e.g. replay malicious traffic on demand)
- Track vulnerabilities across PLC hardware families and firmware releases

Web-view



The screenshot shows the ROO web interface. The top navigation bar includes 'ROO' and the 'Spring' logo. A sidebar menu on the left lists various categories: VULNERABILITIES, BLOCK TYPES, SCANNER, MONITORING SYSTEM, SECURITY TESTS, INSTRUCTION SETS, ATTACK PATTERN, CONNECTION INFO, and IO CONFIGURATIONS. The main content area displays a form titled 'Create new Vulnerabilities' with the following fields:

- Risk Factor:
- Impact Severity Level:
- Affected Services:
- Affected Port:
- Affected Protocols:
- Synopsis:
- Description:
- Possible Solution:
- Plugin Output:

Tomcat Web-Server



Vulnerabilities DB



- Expertise knowledge transfer to Siemens
- ISCI Communication Robustness Tests (CRT)
 - *1st phase*: analysis of the CRT requirements
 - *2nd phase*: Test-bench extension to fulfill the ISCI CRT requirements
 - *3rd phase*: Siemens PLCs testing and eventual vulnerabilities reporting
- Custom S7-protocol and PROFINET security testing
 - Analysis of the proprietary protocols specifications
 - Implementation of S7 and PROFINET specific security tests
- A customized PLC Monitoring System
 - I/O signals process
 - Use of Digital Acquisition Cards
 - Communication Delays
 - Internal device's status
- Multi-Protocols (Man-in-the-middle) layer testing support