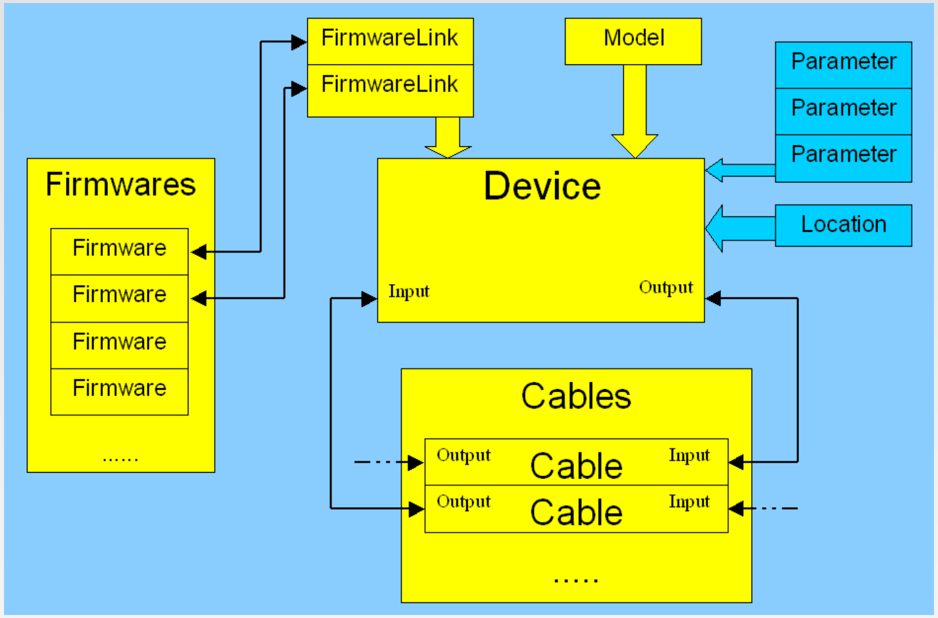
Firmware and Signals database

Bartosz Sakowicz
Piotr Mazur
Marek Kamiński

"Advanced Techniques in LLRF control for XFEL - Collaboration Workshop"
18-20 April 2011

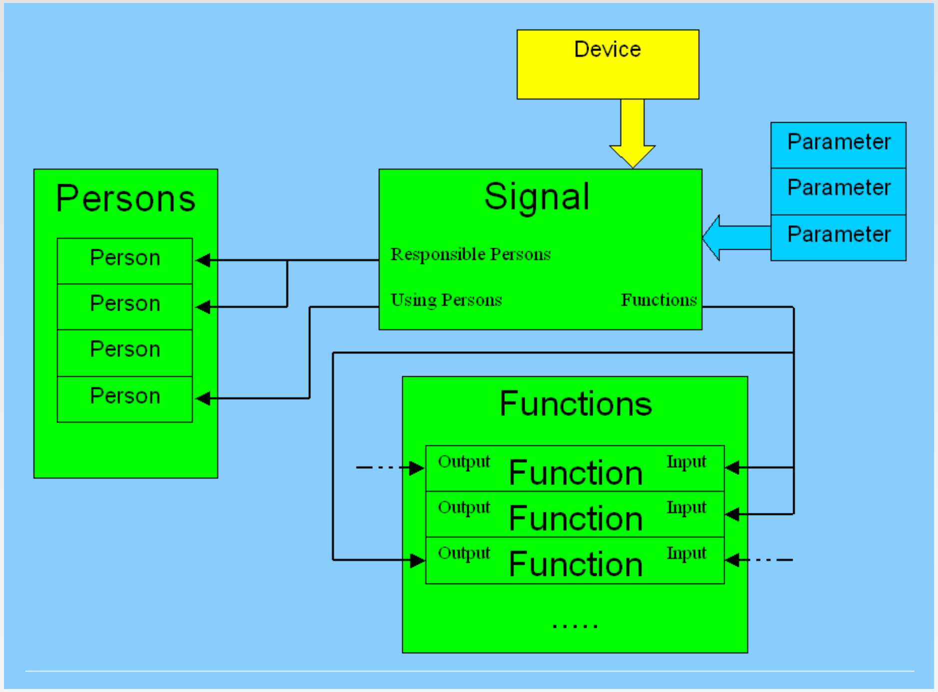
Firmware Database

- **Main goals**
  - Provide a graphical interface to manage structure of the devices that are used in the LLRF system
  - Provide easy to use interface for managing firmware binary and source files
  - Allow to assign correct firmware to a specific device
  - Allow to use the same firmware in several devices distributed across the system
  - Alow for easy testing of experimental firmware files
  - Provide a separate interface for downloading firmware from outside (shell scripts, custom programs etc...)
  - Provide a way to represent cable wirings between devices
  - Allow to specify a set of DOOCS parameters to each node in the tree structure



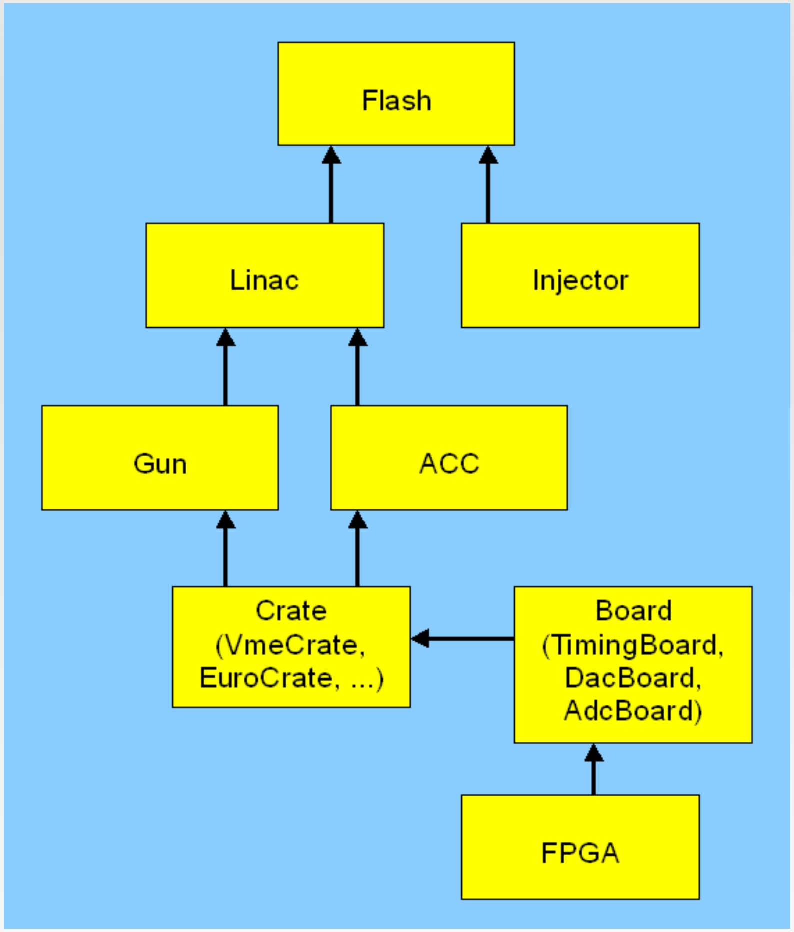
Signals Database

- **Extension to the application providing:**
  - Possibility of adding new nodes: Signals, Functions, Signal Groups and Function Groups
  - Each Function can have multiple input and multiple output signals
  - A signal can be generated by one or more devices allready present in the database
  - A signal can have persons that use that signal and are responsible for that signal
  - A signal can have percent-valued progress
  - A signal has to be used by some function, reporting of signals that are not assigned as an input to a function



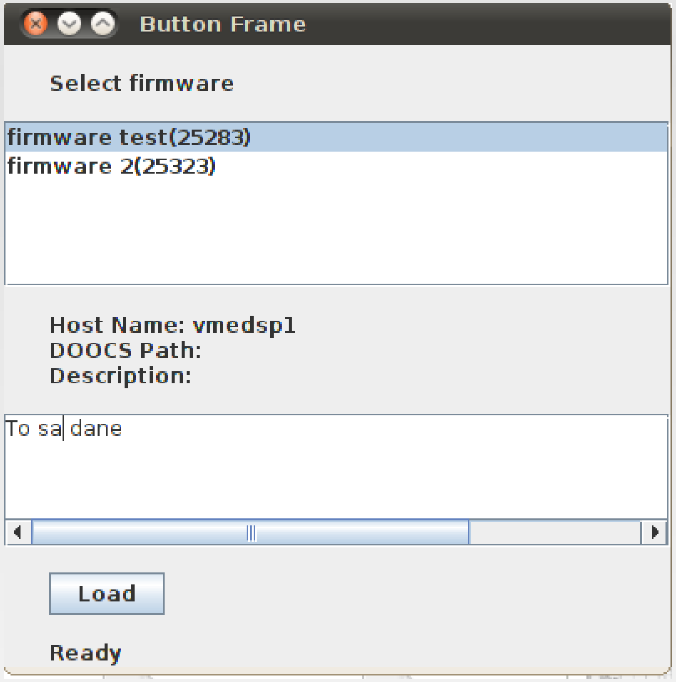
Main system architecture

- The whole application presents the LLRF system as an tree structure
  - The nodes of the tree represent the devices, parameters, firmware files, signals or other elements
  - Most of the nodes cannot be placed in arbitrary place in the tree structure
  - There are nodes which have specific properties:
    - Additional values such as: file attachment, completion status
    - Pointers to other nodes: Firmware nodes, cables etc...
    - Other properties: list of cable inputs and outputs



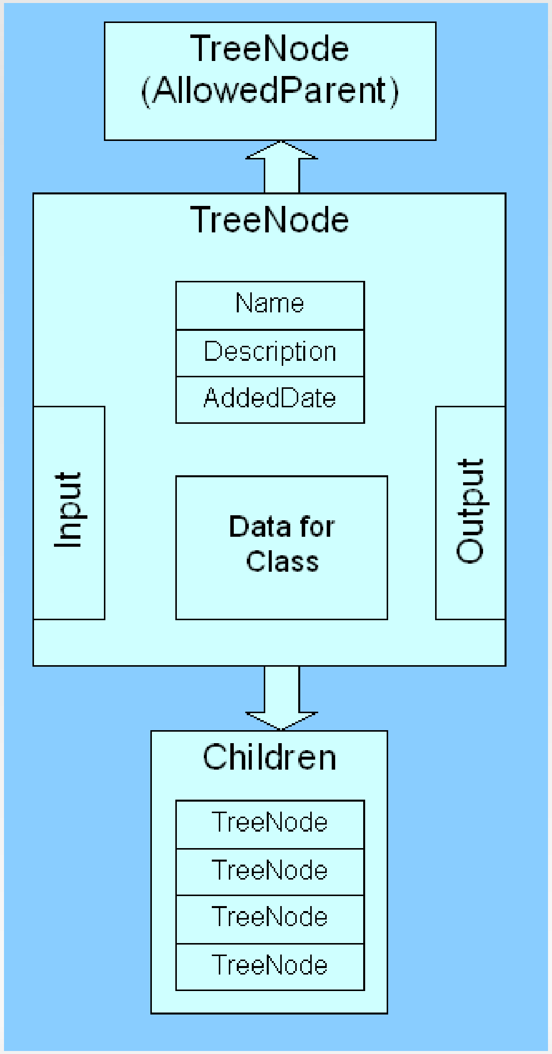
Application design

- Application uses J2EE platform
  - Hibernate ORM framework
  - Struts 2 MVC framework
  - Spring framework as an IoC container
  - Apache CXF for Web Service operations
  - Most of user interface is implemented using AJAX pattern to increase its responsitivity
- Application was deployed on a test server at DESY using Tomcat 6.0 application server and DESY Oracle RAC instance as a backend database
- An example java-based firmware client was deployed and integrated with DOOCS panel



Application design

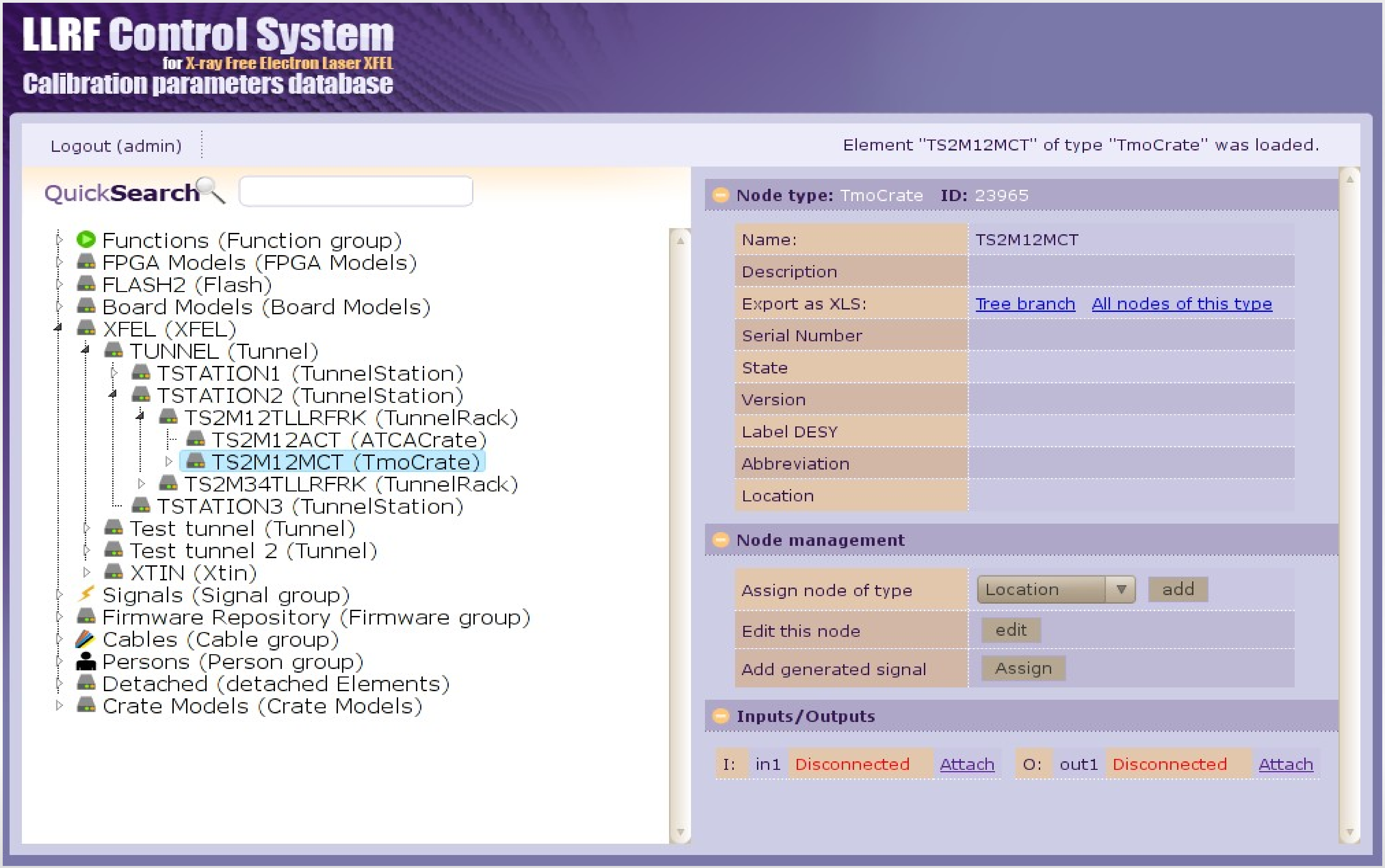
- In order to represent a tree-like structure of different node types an flexible database architecture had to be chosen
- Class-based approach. Each node type in the tree structure is represented by a Java class
  - Natural way of providing similar node types by using class inheritance strategy
  - In addition nodes can have other distinct properties when necesary
  - Structure can be easilly extended by providing new class file, it will be automatically made available to the whole application
- All classes are mapped into a single database table providing significant performance improvement when performing some database operations:
  - Searching for nodes
  - Creating a list of nodes
  - Exporting multiple nodes to **XLS** format



Outcome

- **Outcome:** An application capable of:
  - Representing the structure of an LLRF system in a graphical manner
  - Allowing to quickly find specific device, view its DOOCS parameters, comment the device etc..
  - Allowing to represent cable wirings across devices
  - Allowing for easy firmware management, uploading firmware to the firmware repository and assigning firmware to the specific FPGA device
  - Allowing to represent system completion status via use of Signals, Functions, Signal Groups and Function Groups
  - Allowing to program FPGA devices using firmware located in the database via combination of Web Services and an Java Swing application
    - Device programming can also be performed from command line using a shell script
  - Allowing to export parts of the tree structure in an XLS format

Application screen



Project history

- **2008.12** Idea of firmware database arises
- **2009.03** Formal acceptatance of the task
- **2009.06** Presentation of firmware database in DESY. Signals database concept
- **2009.11** Both databases installed at DESY
- **2010.03** Upgraded versions of databases installed (flash structure implemented)
- **2010.11** Modification of Doocs panel in order to download firmware
- **2011.02** Acceptance of the tasks by Wojtek Jalmuzna, Mariusz Grecki, Holger Schlarb.

The end

Thank you for your attention

Bartosz Sakowicz
Piotr Mazur
Marek Kamiński