XXVth IEEE-SPIE Joint Symposium on Photonics, Web Engineering, Electronics for Astronomy and High Energy Physics Experiments

Virtual Measurement System

Maciej Lipinski

Warszawa, 30.01.2010

Agenda

- ARM and Linux based measurements systems
- The goal
- Software design and its components
- Example applications
- Summary

ARM-based measurement platform

Predecessor: Universal Measurement System with Web Interface:

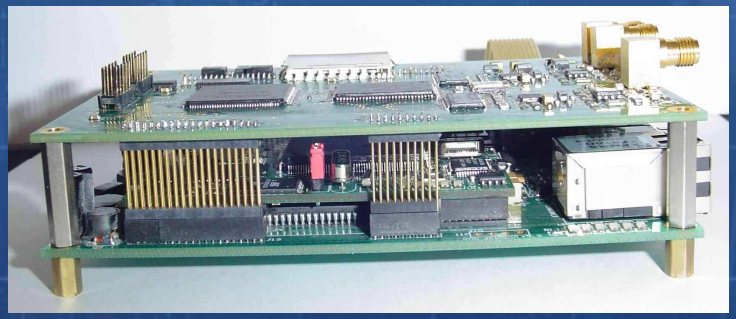
- Powerful basis for implementation of various remotely controlled measurement devices,
- Provides framework for implementation of measurement systems with Web Interface,

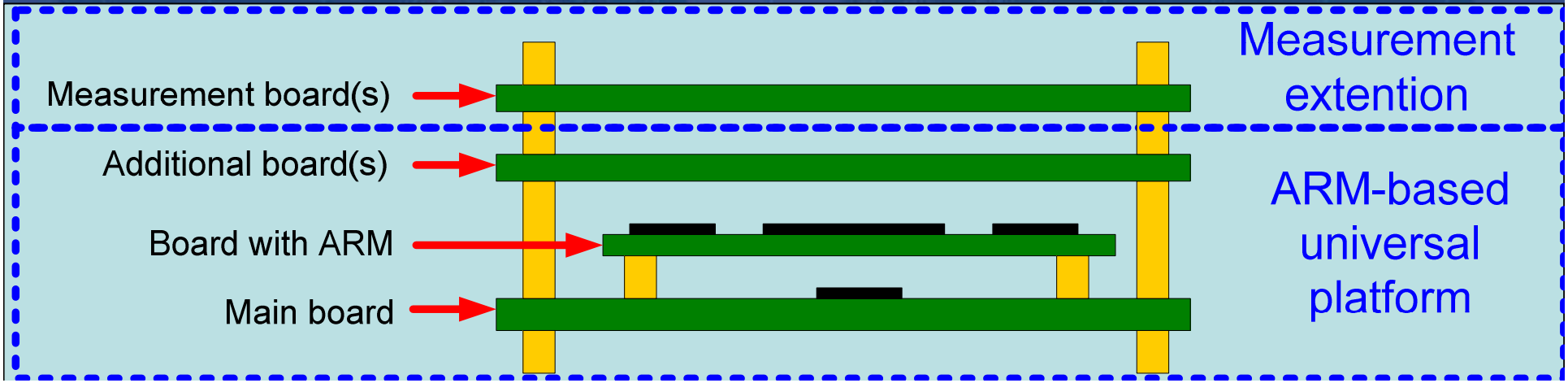
However, development of the system requires:

- Thorough knowledge of the framework
- Good C programming (Linux Device Drivers related) and basic scripting skills

New platform provides additionally:

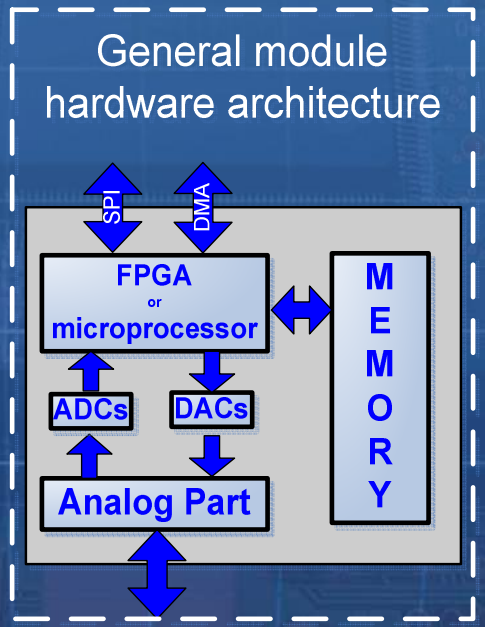
- General Packet Radio Service (GPRS)
- Global Positioning System (GPS)
- Wi–Fi (IEEE 802.11)
- Ethernet
- LCD embedded screen with touchscreen
- Audio input and output

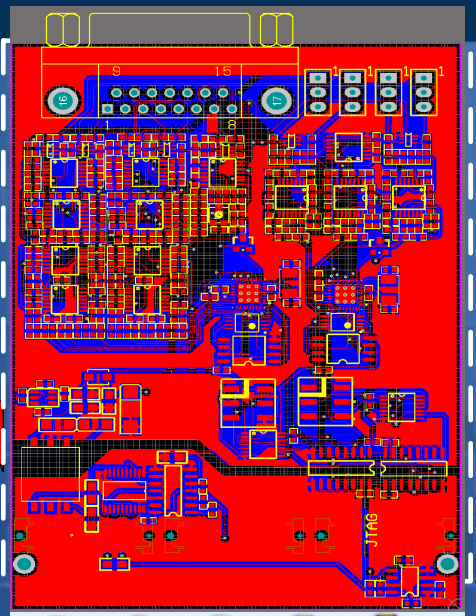


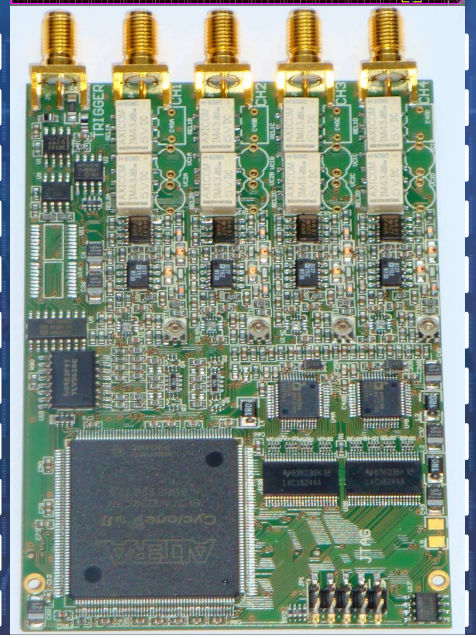


Measurement extensions

- Existing (or under development):
  - 4 channel digital oscilloscope and spectrum analyzer
  - ECG front-end
  - Power quality measurement device
- Prospective:
  - Seismograph front-end
  - Weather station
  - Ultrasound sonar







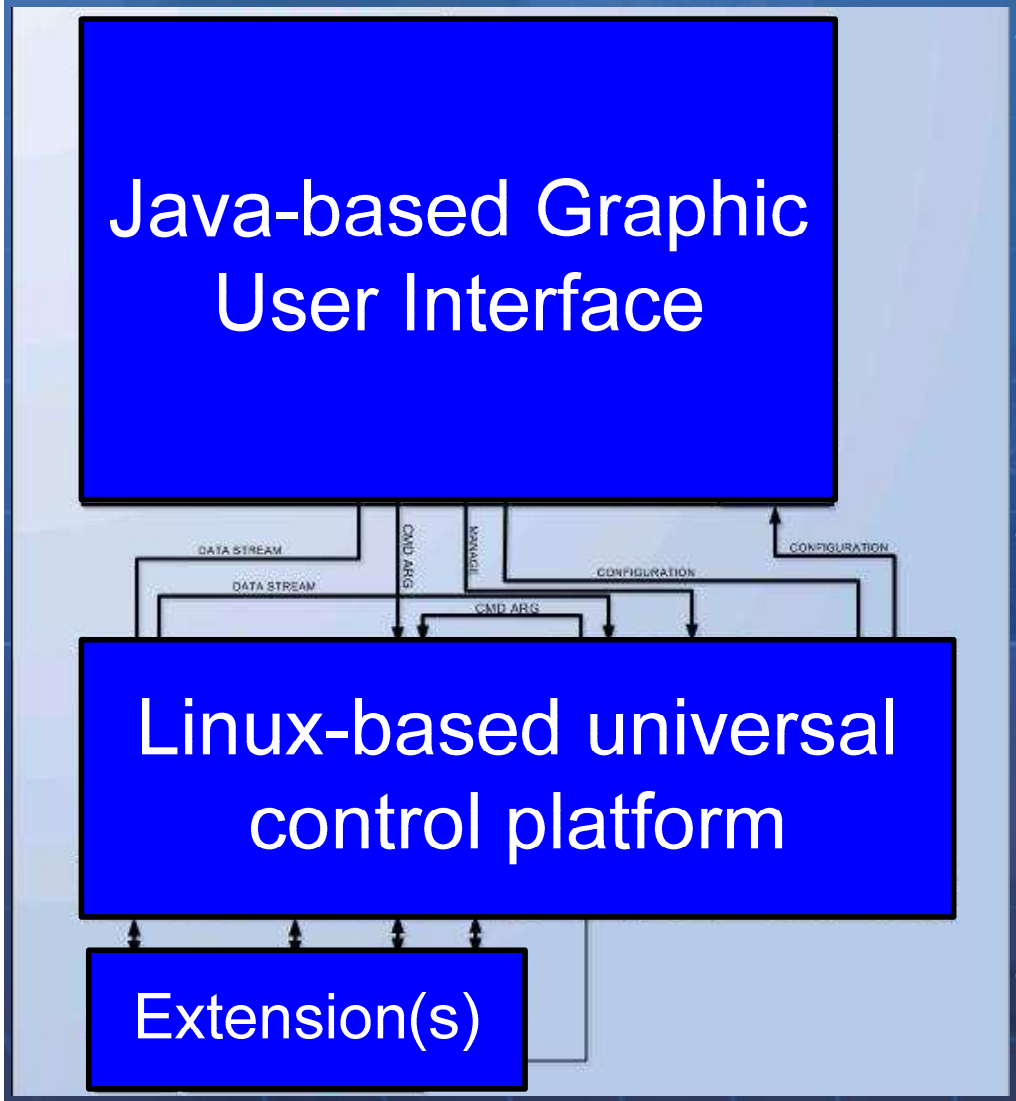
Limited number of forms of communication with ARM-base platform

- Serial Peripheral Interface (SPI)
- Parallel Interface - Static Memory Controller (CMS)

The goal

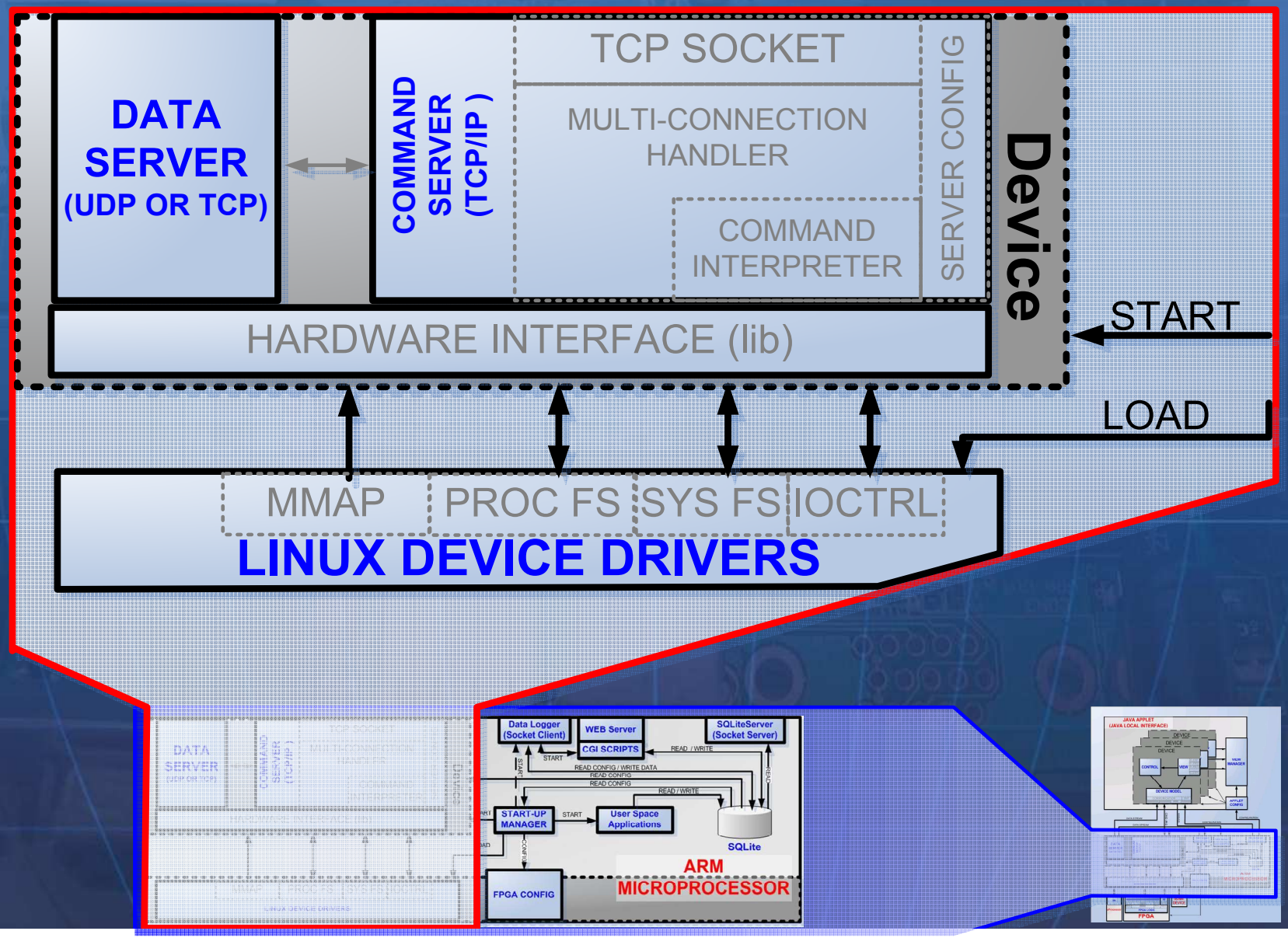
Software design of a virtual measurement system which could enable easy implementation of various measurement instruments based on common Linux-based, reconfigurable control platform.

System's components

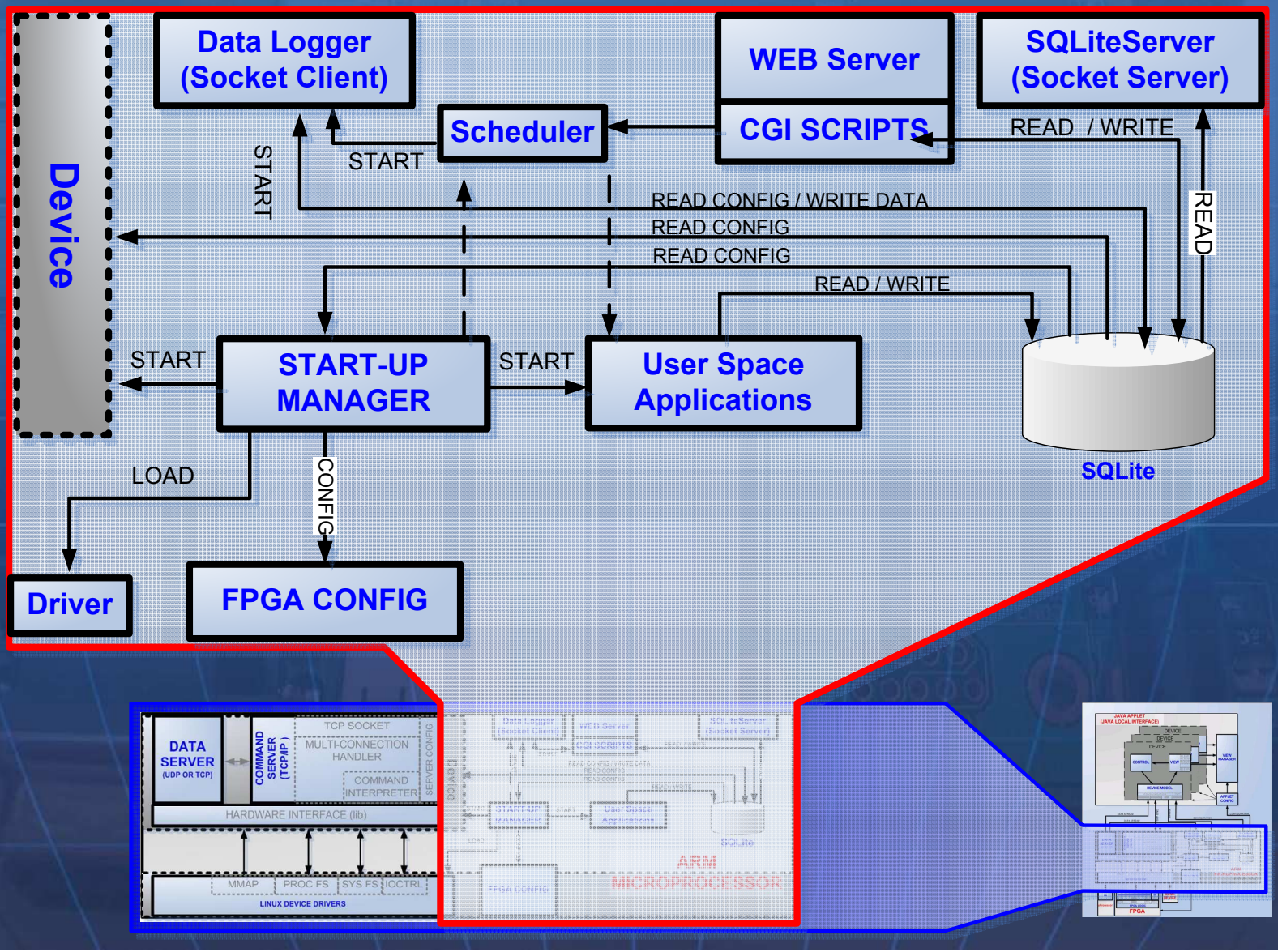


- **Java-based GUI:**
  - Developed for each device
  - Local interface (Java Application)
  - Remote interface (Java Applet)
  - Provides developer with hardware access functions
  - Provides developer with database to store GUI's parameters
- **Embedded Linux-based universal control platform**
  - unified for all measurement extensions
  - based on embedded Linux
  - Configuration stored in Database
  - Transparent to developer
- **Measurement extension(s)**
  - Needs to implement appropriate hardware and software interface with control platform

Linux-based universal control platform (1)



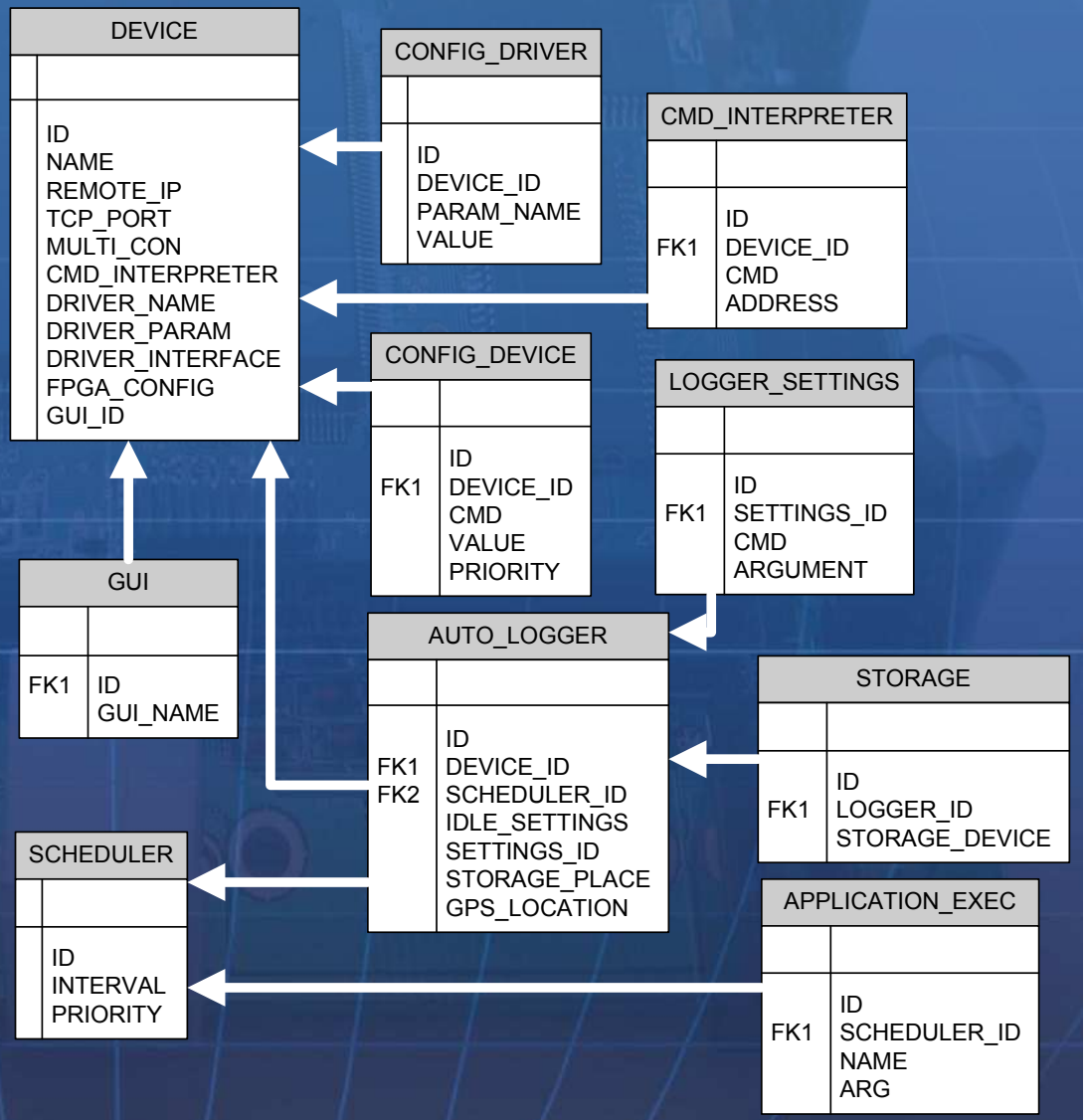
Linux-based universal control platform (2)



Database

(Linux-based universal control platform component)

- The heart of the system
- Enables configuration storage of multiple measurement sets
- After changing measurement extension it is enough to indicate which device is plugged in (automated) for the system to work
- Online administration of the database



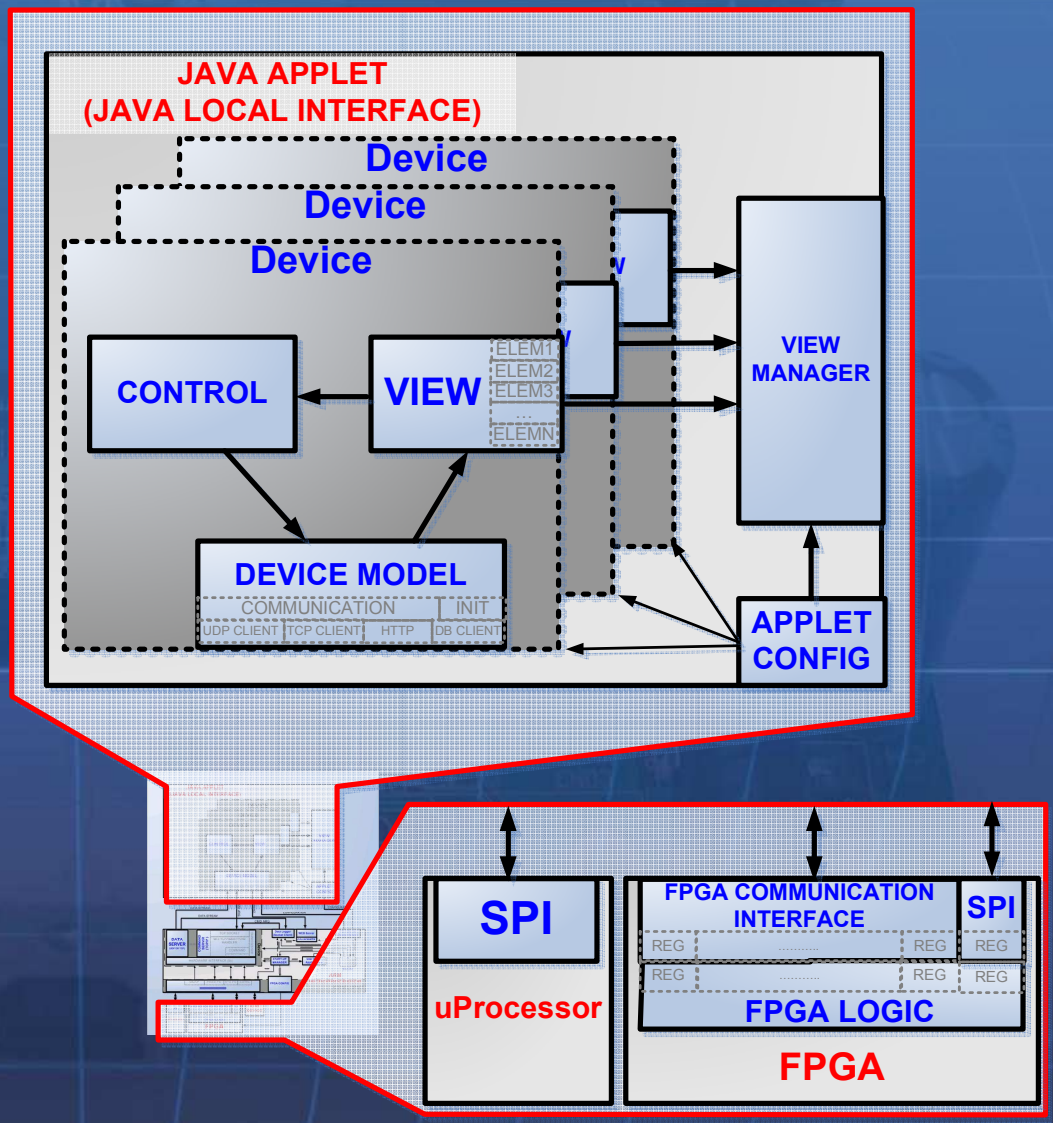
Java-based Graphic User Interface

Graphic User Interface:

- Model-View-Control
- Software-hardware communication provided by framework, its configuration is automatically retrieved from database
- GUI configuration (max/min, parameters) is retrieved from database
- GUI generic elements provided

Measurement Extension:

- The developer of extension needs to implement compatible interface to use universal driver (otherwise, he/she needs to implement custom driver)
- Temples to implement compatible interface (for FPGA, uP) will be provided



Example usage:
ECG front-end implementation

- **Offline Holter configuration:**
  - Data logger periodically stores measurement in database
  - Data can be viewed using GUI
  - Data is uploaded with web interface or copied (it's stored on SD card) to PC for analysis
- **Online Holter configuration**
  - Data logger sends periodically measurements to remote server
  - Data is stored by data logger in database and scheduler periodically runs application which makes measurement analysis and sends alarm through i.e. GPRS
- **Carry-on ECG**:
  - It can be taken by the doctor to the patient
  - It's very small and convenient to carry around
  - The ECG measurement done using local GUI
  - The measurements can be stored (in such case two clients listen to the data server: GUI and data logger)

Summary

- design of highly reconfigurable and
- Easily division between hardware part and software part – imporant if project conducted by
- The development can be divided into several smaller independat projects

Thank you

lipinskimm@gmail.com