

# Split Boot

True network-based booting on heterogeneous MPSoCs

Marvin Fuchs



# Why Split Boot?



Zynq MPSoC

## Processing System

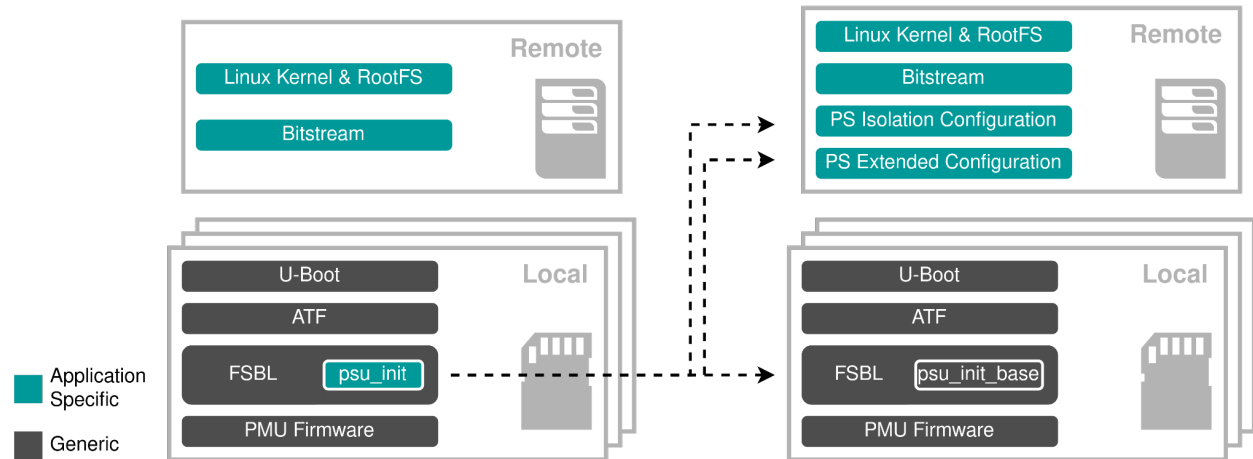
- 4x ARM Cortex-A53
- 2x ARM Cortex-R5

## Programmable Logic

- FPGA

## Challenge:


- Big effort to conventionally **deploy** and **update** a large, distributed system with many Xilinx ZynqMP devices



# What is on our Poster

- Motivation based on the CMS experiment
- The Idea behind Split Boot
- Implementation
- Integration in the development workflow
- Hardware platforms

I am looking forward to answer your Questions in the poster session!



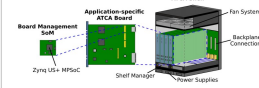
KIT  
Karlsruhe Institute of Technology

## Split Boot

True network-based booting on heterogeneous MPSoCs

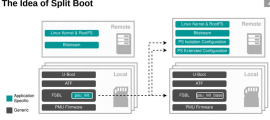
Marvin Fuchs\*, Luis E. Ardila-Perez, Torben Mehner, Oliver Sander  
marvin.fuchs@kit.edu

### Background in Large Scale Experiments (CMS)



- In the context of the High-Luminosity upgrade of the LHC, many custom ATCA electronics boards are being designed, containing Zynq US+ devices for Board management tasks.
- Several hundred Zynq US+ MPSoCs will be used for board management tasks at the CMS experiment alone.
- The traditional approach of flashing updates to each MPSoC individually is not feasible in such a system.
- Our novel proposal is to enable the MPSoCs to fetch all application specific data at runtime from the network themselves.

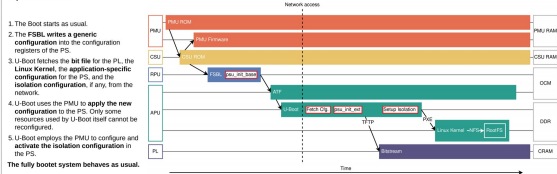
### The Idea of Split Boot



- The approach of Split Boot is to remove all application specific data from the FSBL.
- This allows to have **only generic data stored in the local boot medium**.
- The configuration data for the PS in the FSBL is replaced by a generic configuration to allow booting beyond the FSBL.
- The generic configuration will be replaced with **application specific data** later in the boot process.

All the application-specific information can now be fetched from the network.

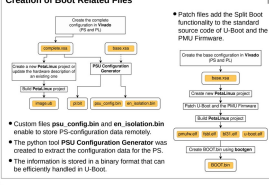
### Implementation




### Conclusion

- The clear separation of generic and application specific data in Split Boot enables to fetch all application specific data at runtime from the network.
- Only a generic base layer of software remains on the local boot medium.
- Split Boot was implemented on the custom Board Management Sub of Serenity-A.
- The mechanism was investigated on Serenity-A in an ATCA shelf to showcase the usability by CMS.
- Split Boot was also tested on three other hardware platforms with two versions of the Xilinx SoC.

### Creation of Boot Related Files



- Custom files `psu_config.bin` and `en_isolation.bin` enable to store PS-configuration data remotely.
- The python tool `PSU Configuration Generator` was created to extract the configuration data for the PS.
- The information is stored in a binary format that can be efficiently handled in U-Boot.



Custom Zynq US+ Board Management Sub

Serenity-A ATCA Board  
(See talk of T. Mehner)

KIT – The Research University in the Helmholtz Association      www.kit.edu