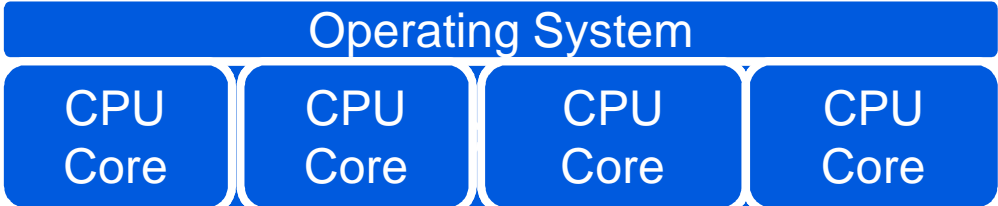
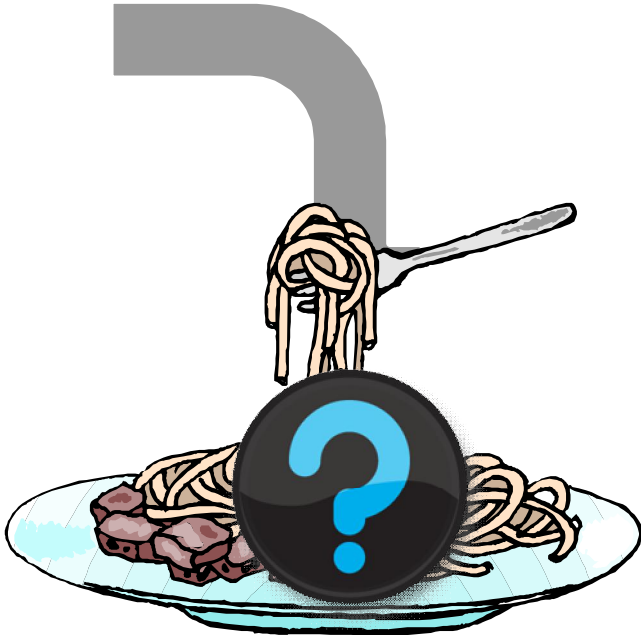
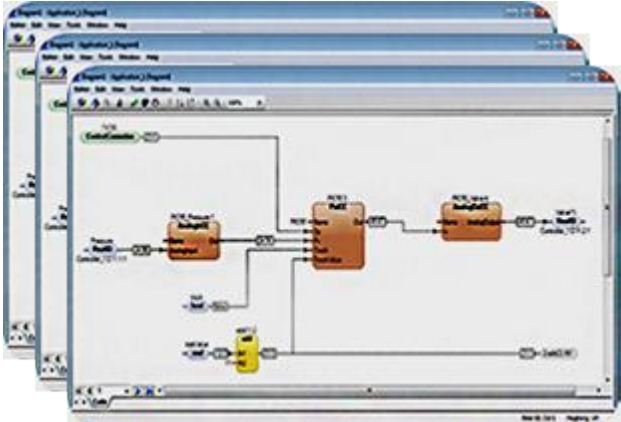




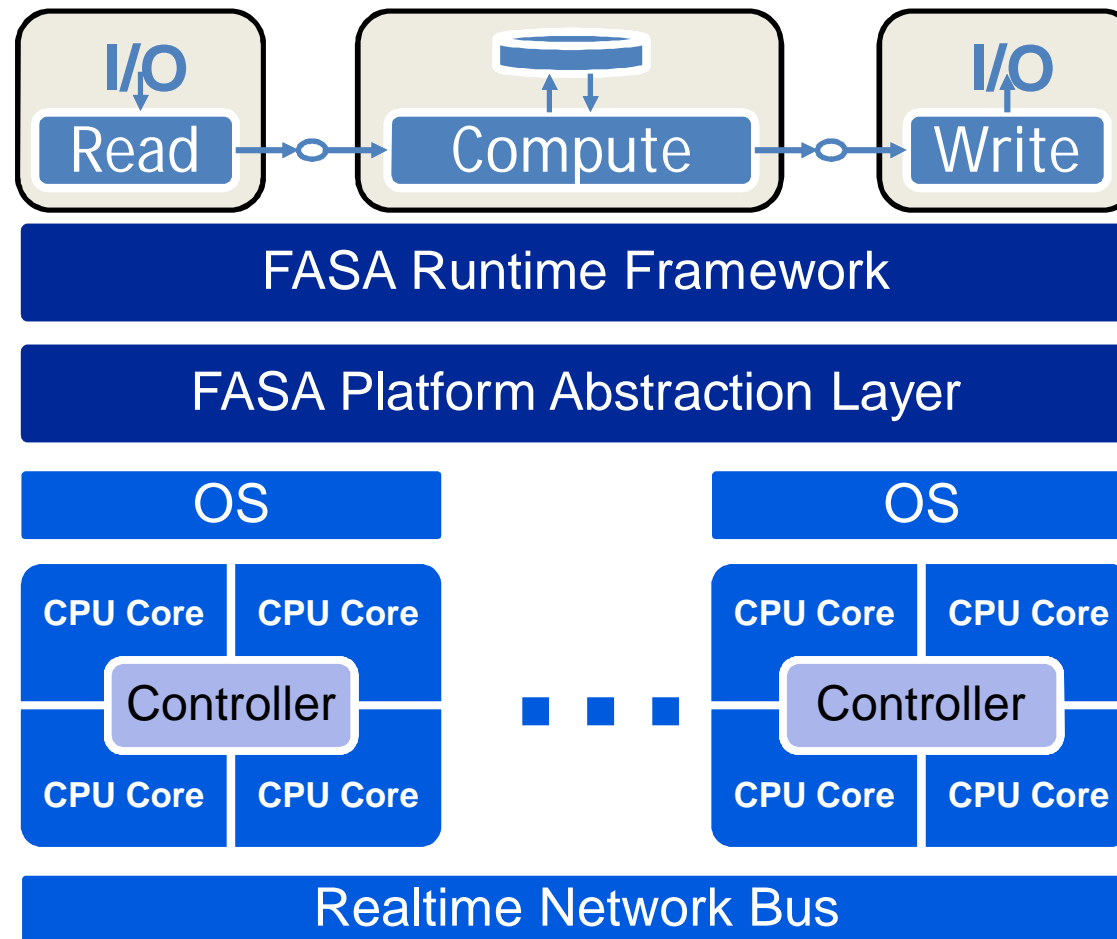
Michael Wahler, Aurélien Monot, Manuel Oriol, ABB Corporate Research, Baden-Dättwil, Switzerland

# Extending 4DIAC with a Partitioning and Scheduling Tool

# Motivation

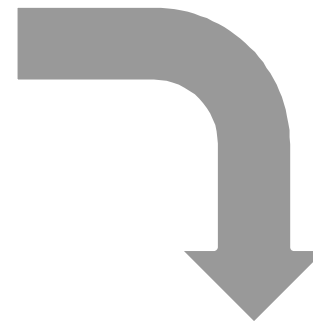
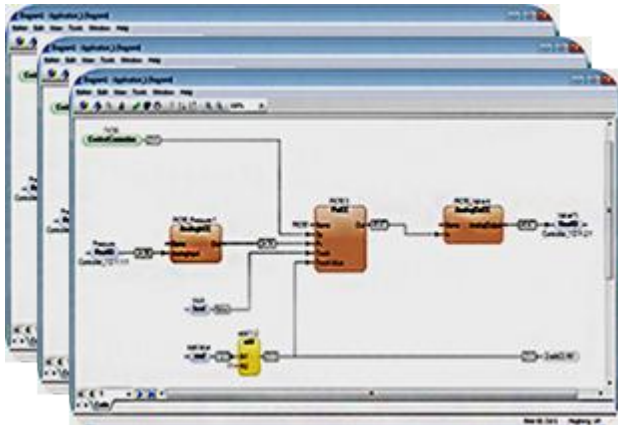


# Target Future Automation System Architecture



# Solution

## Static Allocation and Schedule



### Function blocks → components

- Implementation as C++ objects

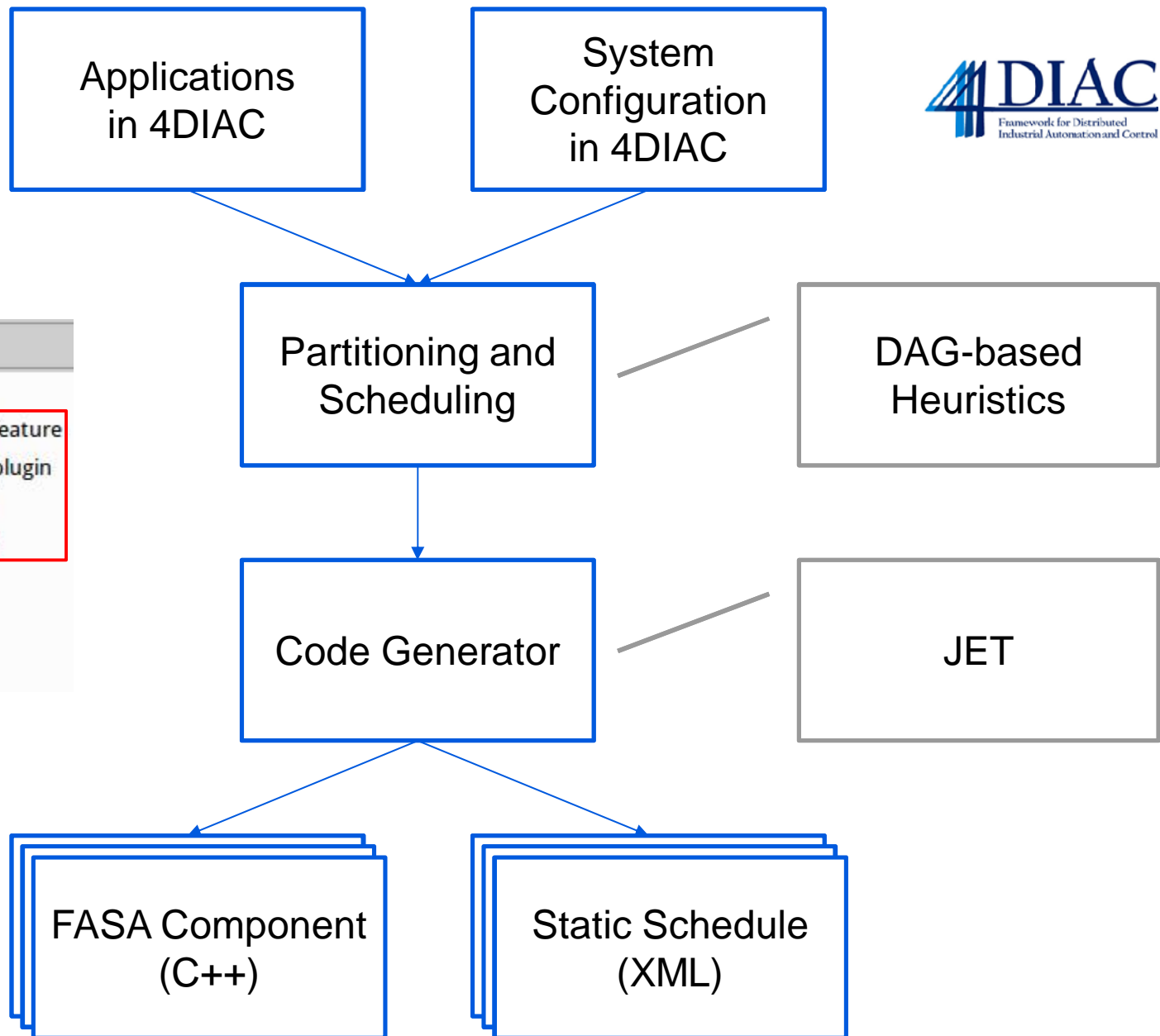
### Static allocation to CPU core

- Reduce cache misses
- Increase predictability

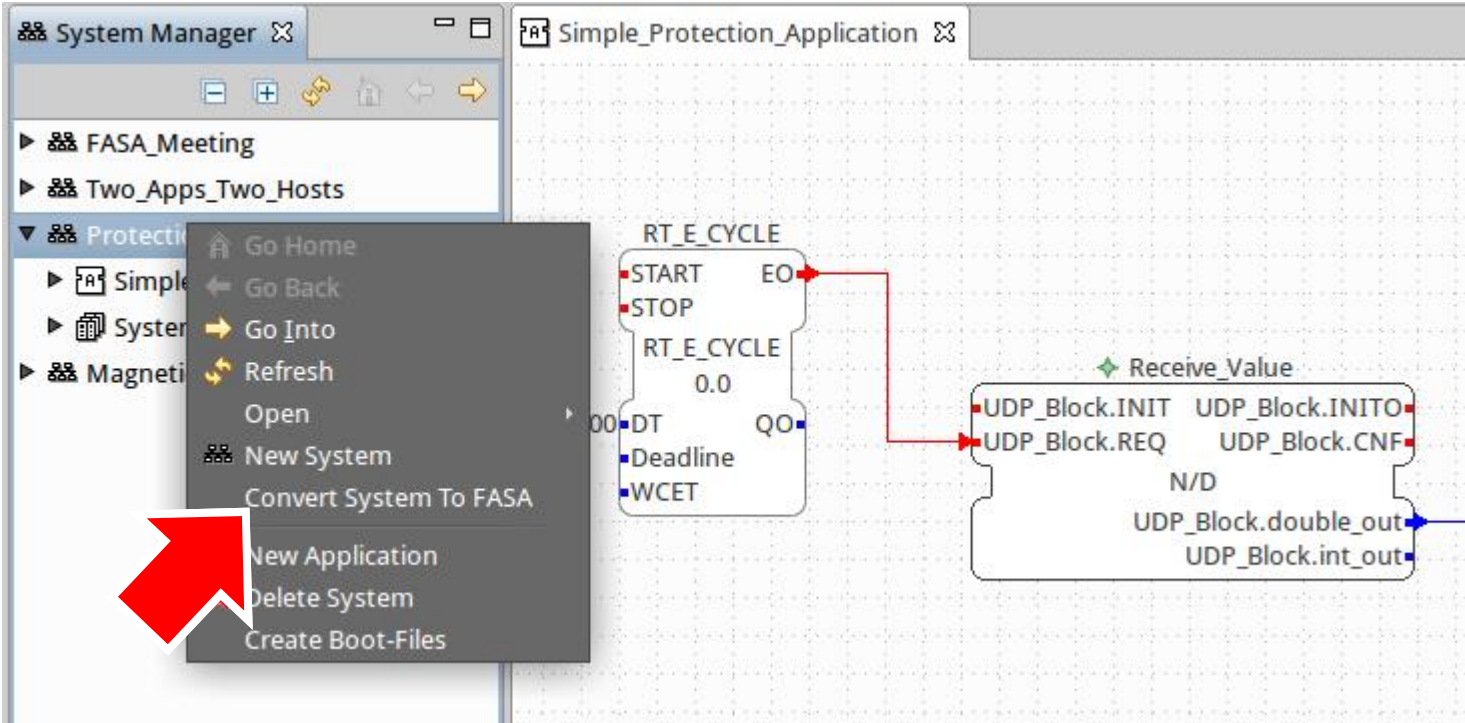
### Static cyclic schedule per core

- Control flow determines execution order

# Details



# Extending 4DIAC Live Demo



# Challenges

- JET sucks
  - But using other transformation engines (e.g., Acceleo) was not possible
- Inconsistencies between model and view in 4DIAC
  - We use the view to complement the model
- Different concepts in 4DIAC and FASA
  - E.g., subapplications vs. components
  - Home-made problem 😊

# Summary

- FASA is a platform for trying out novel research ideas
- 4DIAC is an excellent engineering platform for FASA
  - Based on Eclipse/EMF
  - Mature model editors
  - Easily extensible
  - Friendly and competent support
- Related work at ETFA 2014
  - **Partitioning and scheduling for multi-core**
    - T3.3 (Thursday, Sept 18, 16:50)
  - **Dynamic software updates with FASA**
    - T1.6 (Thursday, Sept 18, 15:00)
  - **Efficient M-out-of-N Fault Tolerance**
    - SS03.1 (Wednesday, Sept 17, 16:50)



Power and productivity  
for a better world™

