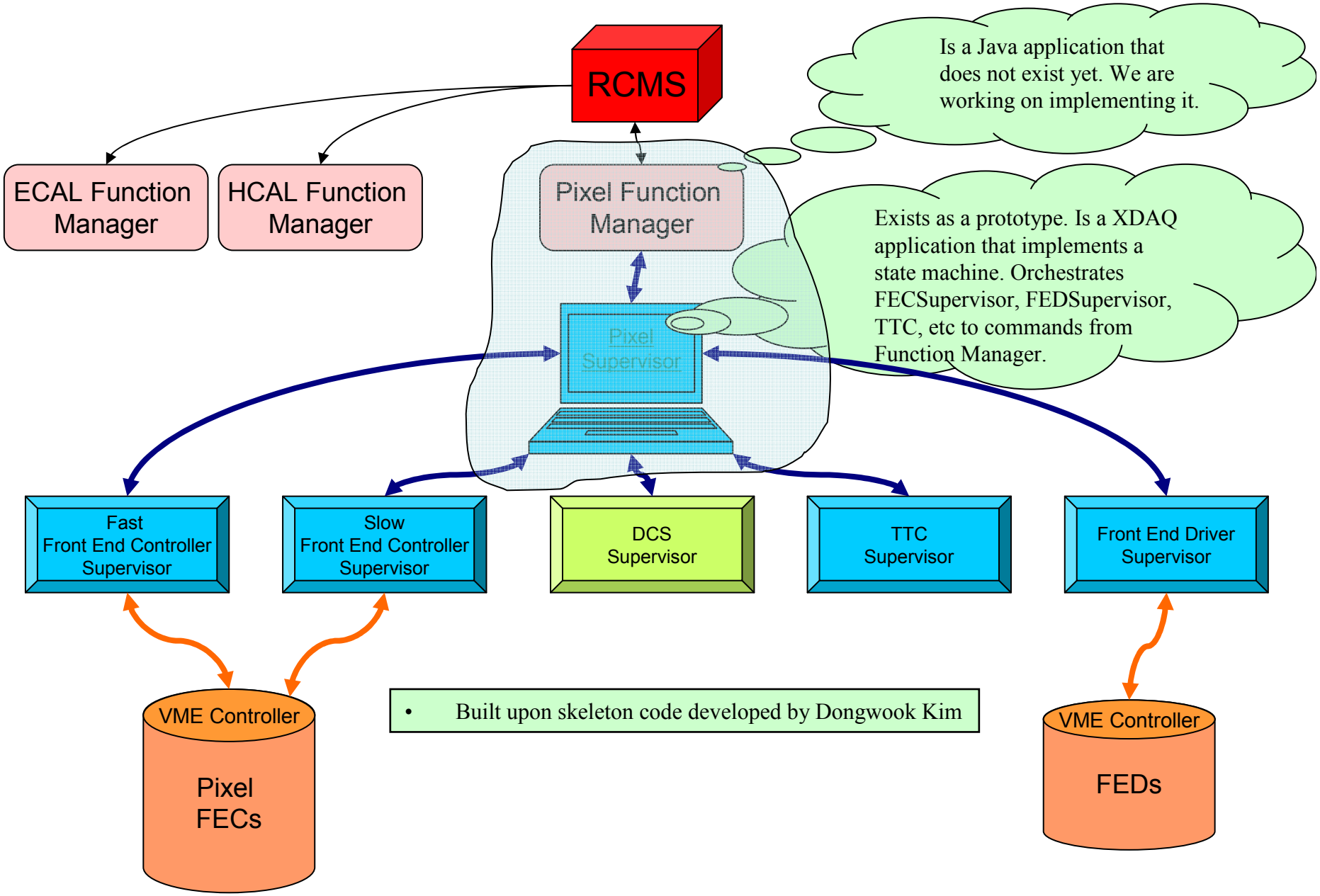
Pixel Run Control and Calibration

-Souvik Das
under Anders Ryd and Karl Ecklund

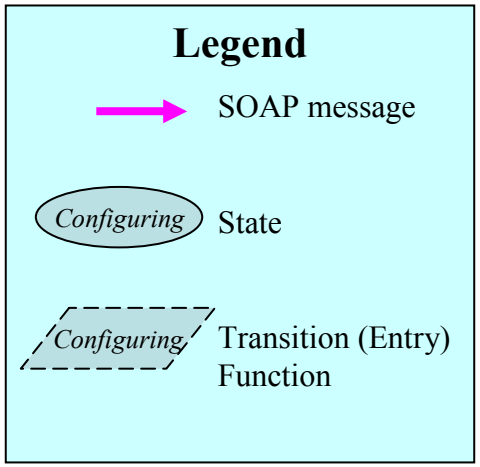
Where Am I Working?

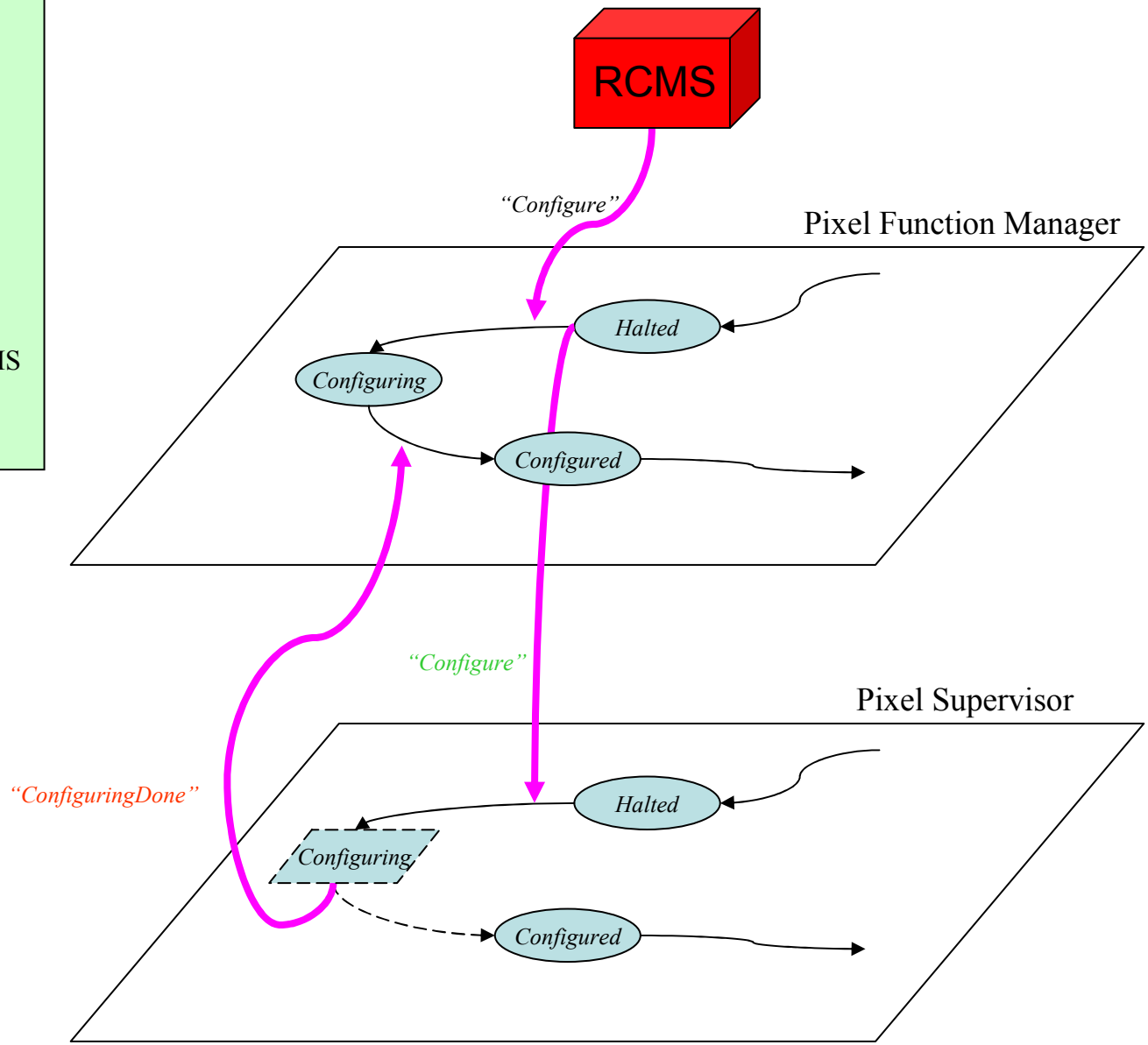


State Machine Layers

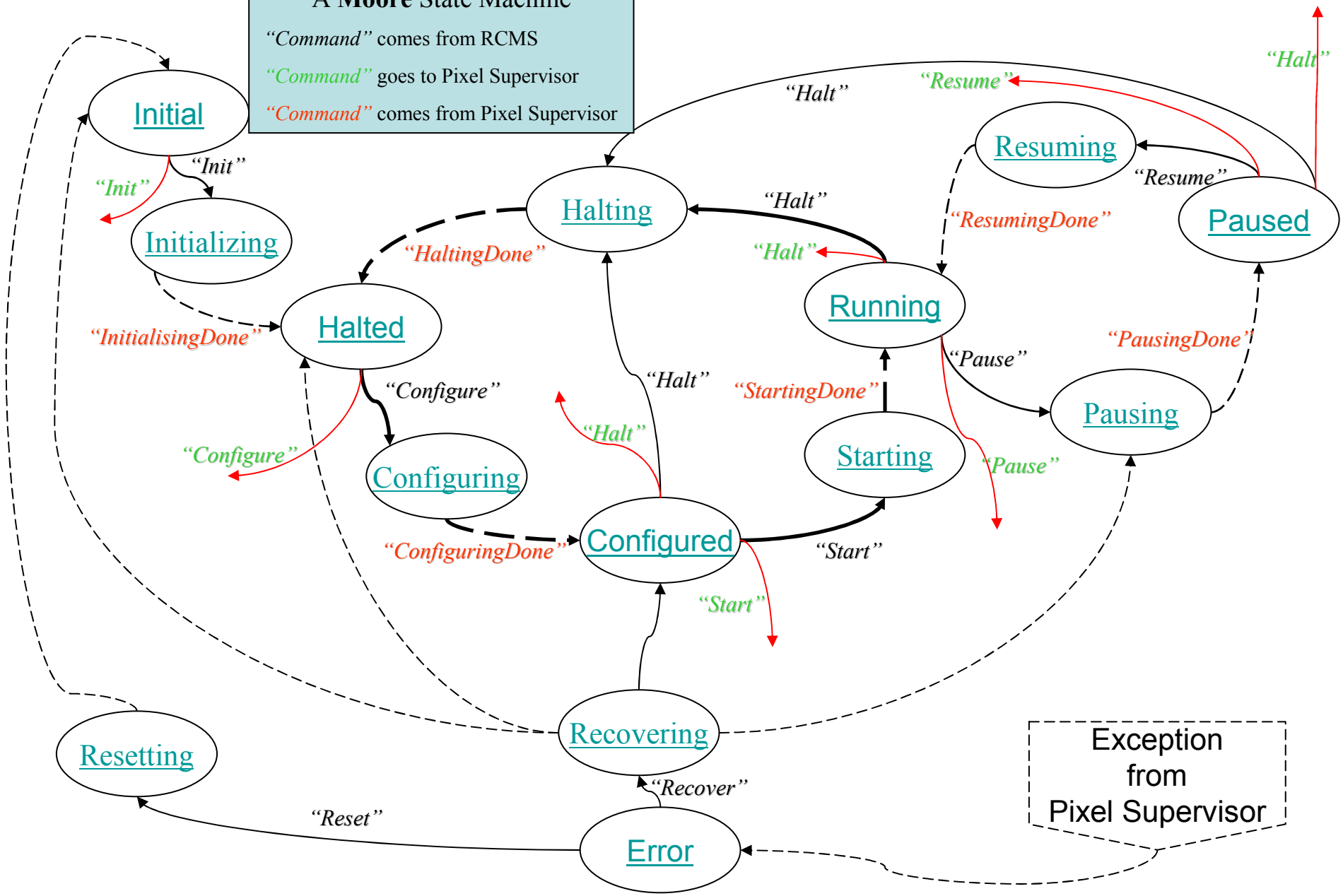
Why two layers?

- Level 1 Function Manager (Version 1.0) proposed by Alexander Oh is a Moore State Machine.
- Pixel Supervisor has no reason to be Moore. Mealy is economic on states and easily implemented in XDAQ. (Alex Oh mentions this in email.)
- Definite states of Function Manager ensures that status queries from RCMS will be always answered.

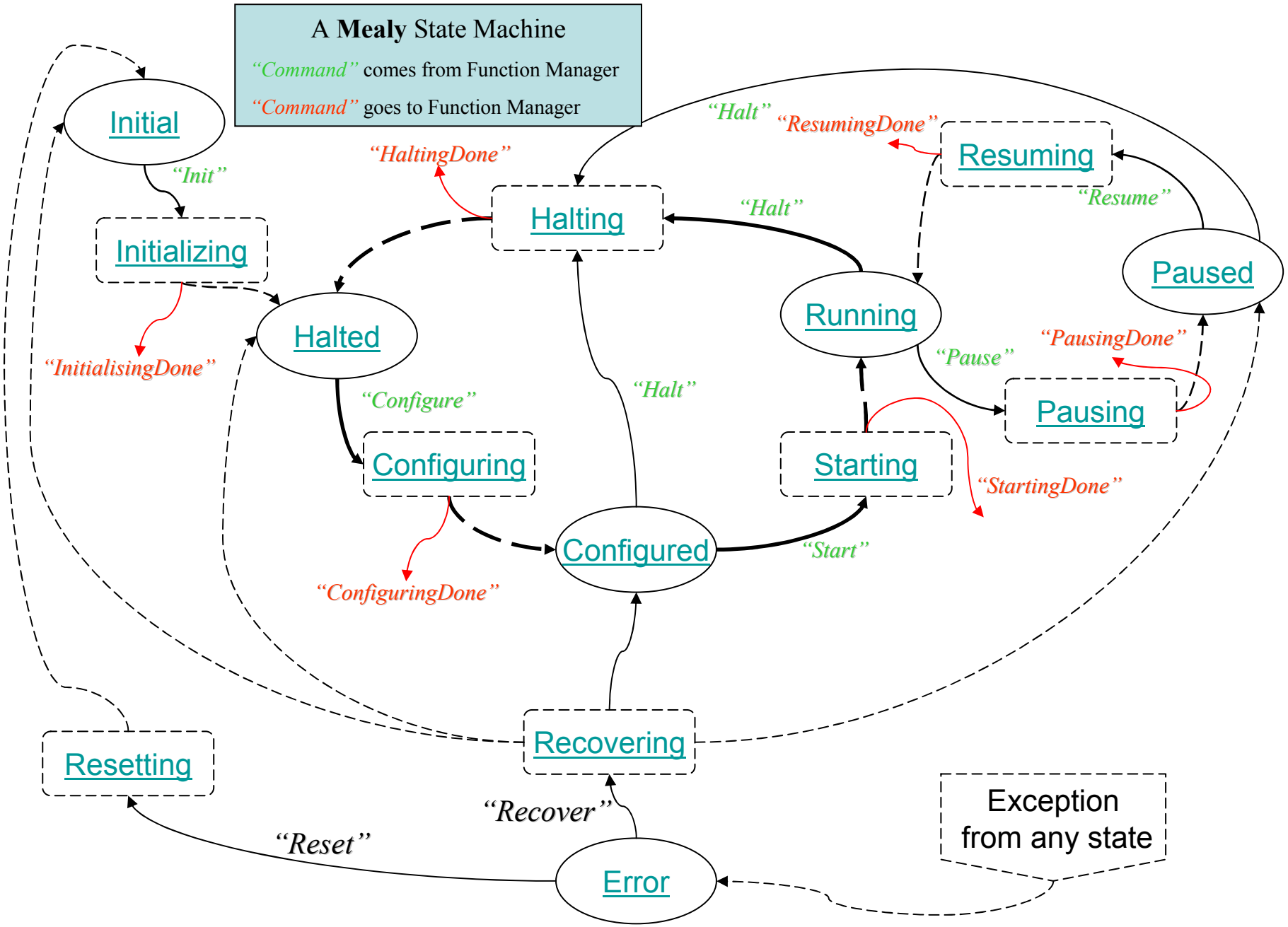




Function Manager (Web State Machine for now)



Pixel Supervisor



Pixel Supervisor

Recent Changes

- A working prototype exists as a XDAQ application and may be checked out from the /TriDAS/pixel/PixelSupervisor area of our CVS repository. (It has been moved from its /TriDAS/pixel/SkeletonPixelDaq/PixelSupervisor area.)
- Receives a configuration key with the “Configure” command to switch between Physics and Calibration runs. The option appears as a radio button on the GUI.
- Implemented a dummy Gain Calibration Algorithm within the “Running” state.

To Do

- Need to interface with the Database Supervisor for configuration requirements.
- S-Curve and other calibration algorithms have yet to find their place.
- The “Recover” commands and state have not been implemented yet.

(Pixel) FEC Supervisor

Recent Changes

- A prototype exists as a XDAQ application and may be checked out from the /TriDAS/pixel/FECSupervisor area of our CVS repository. (It has been moved from its /TriDAS/pixel/SkeletonPixelDaq/FECSupervisor area.)
- FEC Supervisor has two types of commands that may be issued to it:
  - •High Level Commands to implement state transitions
    - •Init
    - •Configure
    - •Start
    - •Pause
    - •Resume
    - •Halt
  - •Low Level Commands for calibration algorithms. Extended versions of the PS146 Pixel Chip programming commands.
    - •Prog_DAC (VME Address, Hub Address, Port ID, RoC ID, DAC Address, DAC Value)
    - •Prog_Pix (VME Address, Hub Address, Port ID, RoC ID, Column, Row, Data byte)
    - •Cal_Pix (VME Address, Hub Address, Port ID, RoC ID, Column, Row, Data byte)
    - •ClrCal (VME Address, Hub Address, Port ID, RoC ID)
    - •The first four arguments of each command can either contain values or **a wildcard "*".** A wildcard means **the command must be sent down to all possible argument values**.

To Do

- We need to get some idea of what must be done in each state. Refine and add to interfaces by working through calibration and other procedures.

LTC Supervisor

Done

- Our newly created LTC Supervisor is not a state machine. All its methods are placeholders. It can accept the following commands via SOAP:
  - •Reset
  - •Configure (Key)
  - •Enable
  - •Pause
  - •Resume
  - •Halt
  - •Trigger (Number)
  - •ReadCounters
  - •ResetCounters

To Do

- Refine and add to interfaces.

A Gain Calibration Algorithm

Pseudo Code Implemented in PixelSupervisor's "Running" State

Pick a *VME Address*, *Hub Address*, *Port ID* and *RoC ID* for calibration. Any of them can be a *"*"*.

Specify two ends of the rectangular mask of pixels on the chip you want to calibrate.

For (*Column*, *Row* efficiently scanning the mask) // Move over diagonals for efficiency.

{

To FECSupervisor send Cal_Pix*(VME Address*, *Hub Address*, *Port ID*, *RoC ID*, *Column*, *Row*, *"0x01"*)

}

To FEDSupervisor send "Enable SpyFIFO Line"

For (*VCal* from 0 to 255)

{

To FECSupervisor send Prog_DAC(*VME Address*, *Hub Address*, *Port ID*, *RoC ID*, *"0x10"*, *VCal*)

To LTCSupervisor send Trigger(??)

To FEDSupervisor send "Read SpyFIFO Line"

}

To FEDSupervisor send "Disable SpyFIFO Line"

To FECSupervisor send ClrCal(*VME Address*, *Hub Address*, *Port ID*, *RoC ID*)

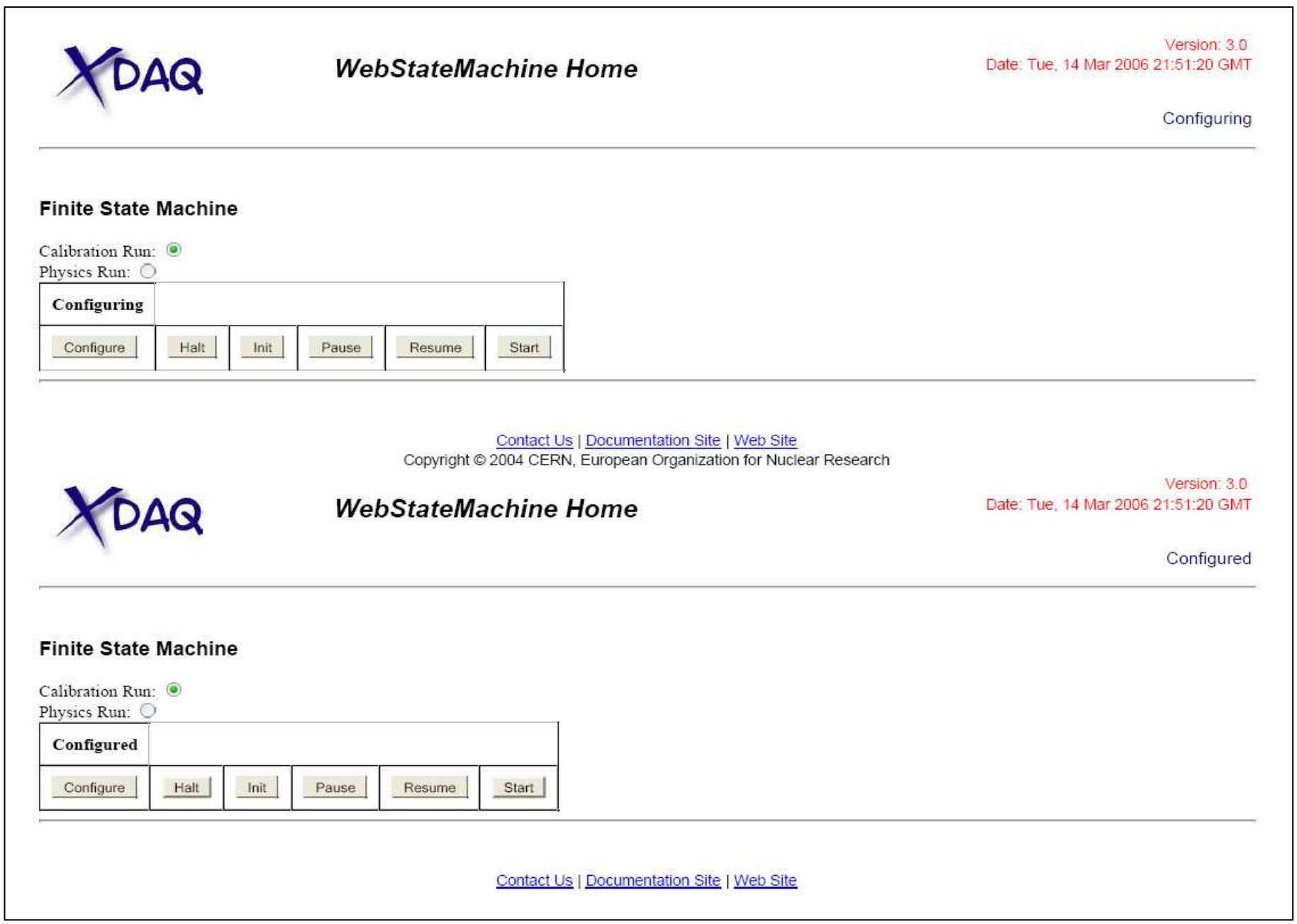
Time for one SOAP Transaction

- Between XDAQ applications on the same CPU, same Port: $(2.31 \pm 0.03) \times 10^{-4}$ sec
- Between XDAQ applications on the same CPU, different Ports: $(1.05 \pm 0.06) \times 10^{-2}$ sec
- Between XDAQ applications on different CPU connected via 11 Mbps Wireless Network: sec

Summary and To Do List

- Evolved the skeleton code of Dongwook Kim and checked them into the TriDAS/pixel area of our CVS repository. Makefiles are also available for each project to facilitate building. A profile.xml file has been provided for automatic uploading of the various applications into XDAQ.
- We have also begun looking at the RCMS Function Manager code written in Java to see how it fits in with our Pixel Supervisor. The states of Pixel Supervisor (, FEC Supervisor and FED Supervisor) should be designed as much in alignment with the Pixel Function Manager states as possible.
- We have put in hooks for low level commands necessary for our dummy calibration algorithm in the FEC Supervisor and the LTC Supervisor.
- Having implemented a dummy calibration algorithm, we realise that we need to have some interface with the FED Supervisor that allows us to **enable, disable and read the SpyFIFO line**.
- Having read the SpyFIFO line, we need to decide on how we must channel the data for processing and how we must we must respond to the results of the processing. That is, close the calibration loop.
- Prod on with the SOAP timing tests to see how network topologies affect it and if we could run into possible bottlenecks. The wildcard * was introduced with such SOAPy situations in mind.
- Work in closer alliance with hardware to make sure our software actually works with it. We do not have FEC / FED crates at Cornell and are making software in a vacuum.
- Updates, notes, questions etc will be posted on the HyperNews forum: https://hypernews.cern.ch/HyperNews/CMS/get/pixelOnlineSW.html

The GUI


XDAQ
WebStateMachine Home
Version: 3.0
Date: Tue, 14 Mar 2006 21:51:20 GMT
Configuring
Finite State Machine
Calibration Run:
Physics Run:
Configuring
Configure
Halt
Init
Pause
Resume
Start
Contact Us | Documentation Site | Web Site
Copyright © 2004 CERN, European Organization for Nuclear Research
XDAQ
WebStateMachine Home
Version: 3.0
Date: Tue, 14 Mar 2006 21:51:20 GMT
Configured
Finite State Machine
Calibration Run:
Physics Run:
Configured
Configure
Halt
Init
Pause
Resume
Start
Contact Us | Documentation Site | Web Site