Common Concerns at Architectural Level

• ACS is Multi-processor, hard real-time, distributed system. ACS and EDCS have shared concerns (specification, decision making)
  – FPGA’s are components in system
  – Need to represent and analyse at multiple timescales & granualities
  – Need multiple application specific notations accommodating user’s development paradigm
ACS Needs a “Reconfiguration Manager”

• Reconfiguration Specification
  – Trigger events and detection mechanisms
    • State transition diagrams
    • Temporal sequences
    • QoS
  – Cost functions
  – State to be transferred
  – Synchronization
  – Transferring State (control & code)
Component composition (another view of Reconfiguration Manager)

- More complex systems will require evidence of safe composition
- Dynamic architectures pose problem of safe switching (Reconfiguration Manager)
  - expressing preconditions (and postconditions ?)
  - saving required state
  - similar to validating mode switching in software avionics
Specification techniques, representations, notations

- Expressing dynamic aspects
- Domain (and implementation) specific styles
- Multi-style capability
- Event and state representation (and management)
Validation/Verification

- Computational correctness
- Hard real-time constraints
- Standard deadlock, livelock concerns (FPGA as system resource)
- Formal methods applied to ACS design
- Value of EDCS architecture centered T&A technology for reconfigurable hardware?
Partitioning Issues

• With respect to functionality, hardware type and time
• Needs to consider storage and control operators and hardware
  – GPP, DSP, FPGA, ASIC, etc.
General design Issues

- Ability to work at a higher level of abstraction
- Design negotiation/collaboration
- Tools for design space exploration (multiple levels of precision/granularity)
  - power/thermal
  - real estate
  - computational latency
  - reconfiguration latency
  - life cycle costs
- Seamless migration from concept -> picture/diagram -> code -> HW/SW partition -> execution