



# Towards Architecture-based Self-Healing Systems

Eric M. Dashofy, André van der Hoek,  
and Richard N. Taylor

WOSS'02

November 18, 2002



# What is “self-healing?”

Key Question: What is the difference between a fault-tolerant and a self-healing system?

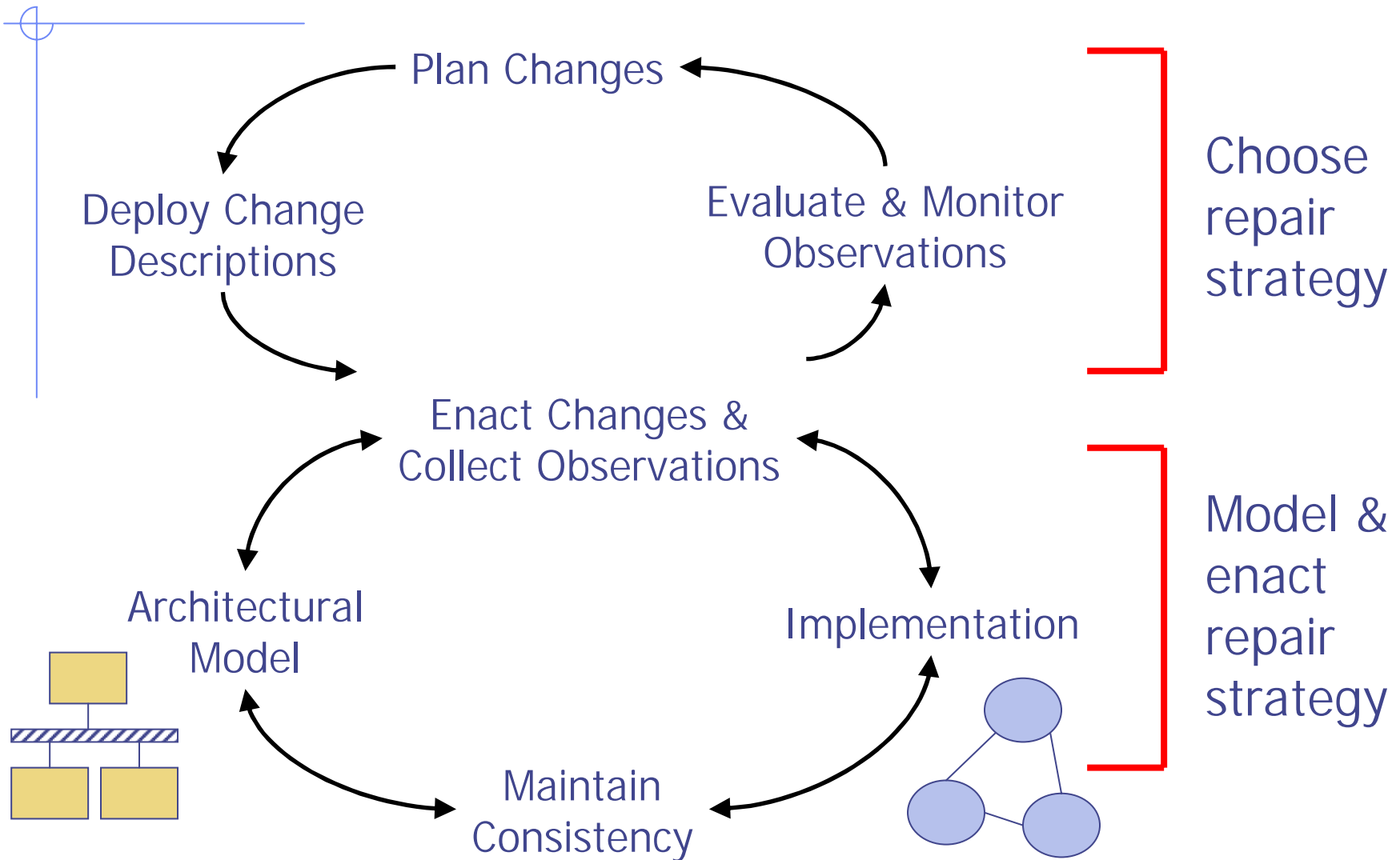
## ◆ Fault-Tolerant

- n Connotes fault-based repair and understanding
- n Faults are likely pre-specified
- n Repair strategies are also pre-specified

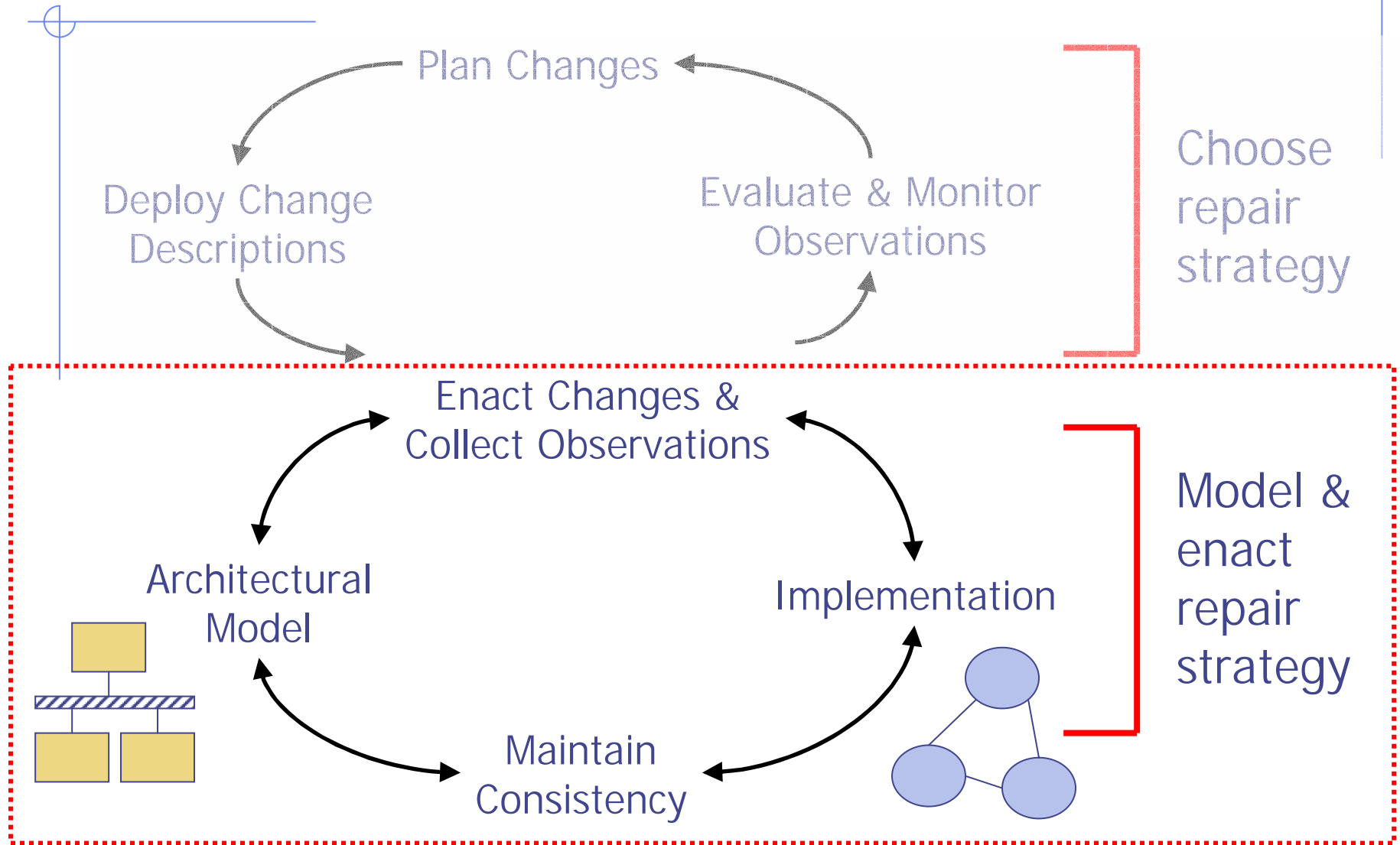
## ◆ Self-Healing

- n Connotes goal-based repair and understanding
- n Unexpected faults are expected
- n Arbitrary repair strategies constructed at runtime

# Overall Vision



# Our Focus



# Additional Aspects of the Approach

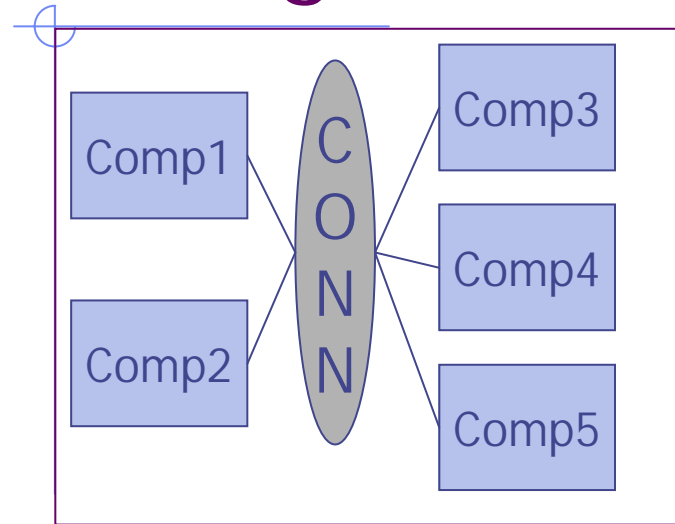
## ◆ Architectural Styles

- n Loosely-coupled, event-based
- n Foundation for runtime change
- n Foundation for monitoring

## ◆ Systems described in extensible ADL

- n Description accompanies deployed system
- n Repair strategies expressed in terms of architecture description

# Expressing Repair Strategies Using Architecture Differencing



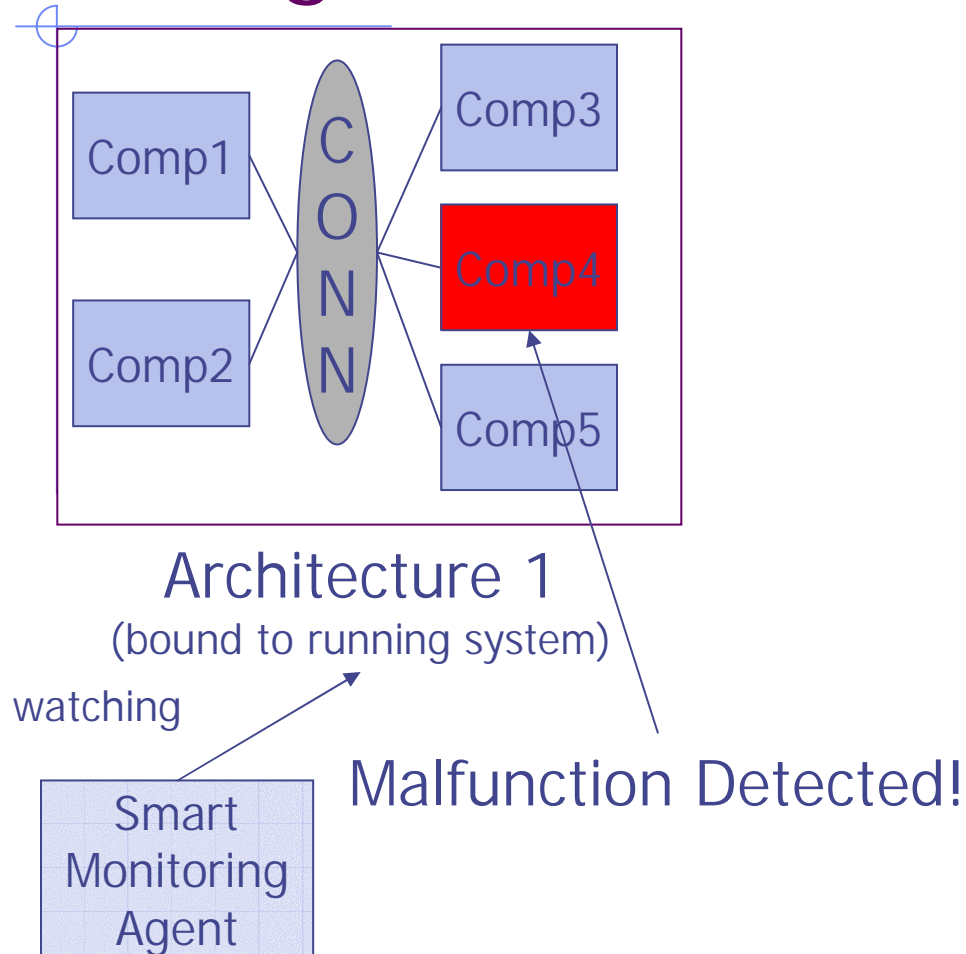
Architecture 1

(bound to running system)

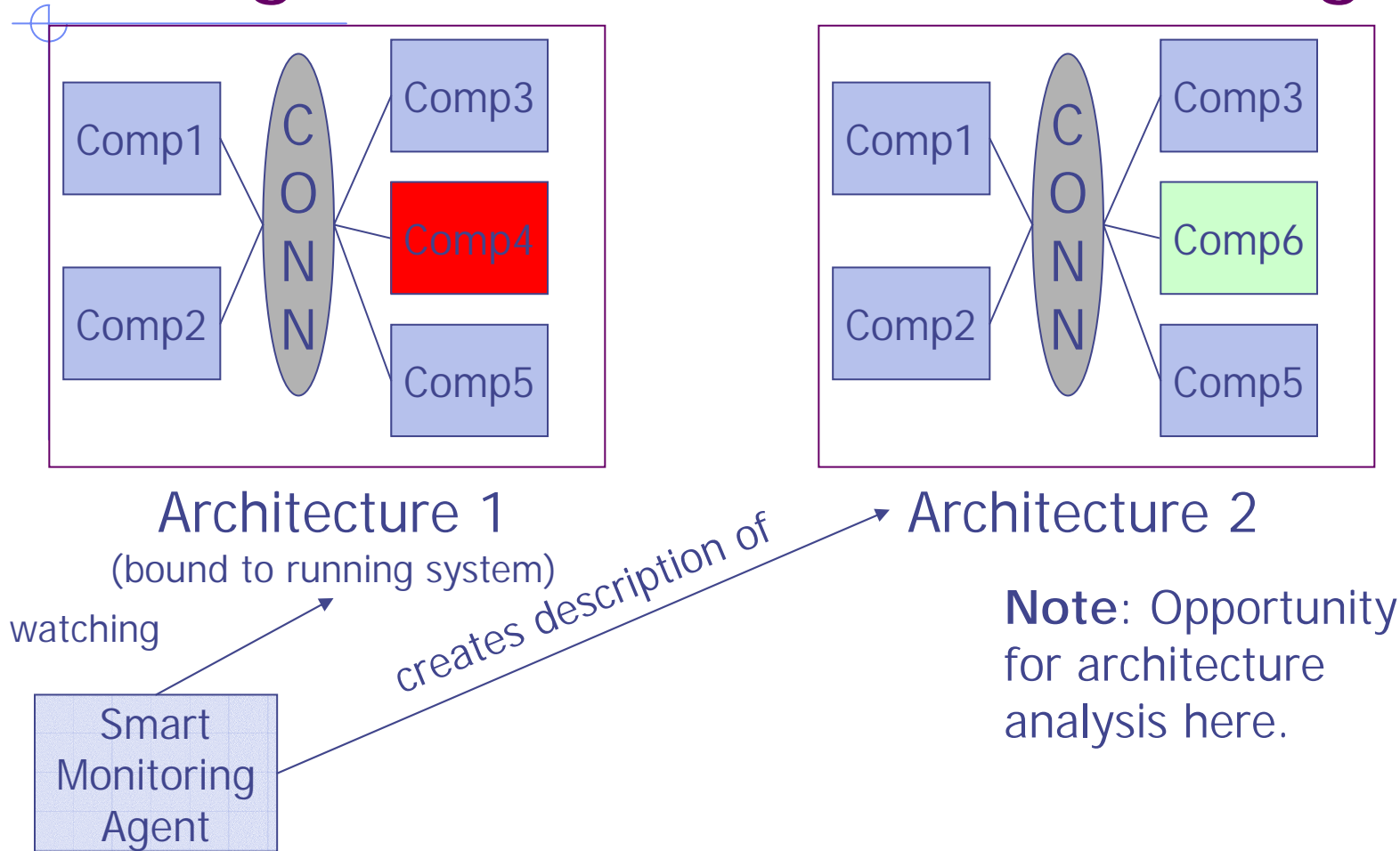
watching



# Expressing Repair Strategies Using Architecture Differencing

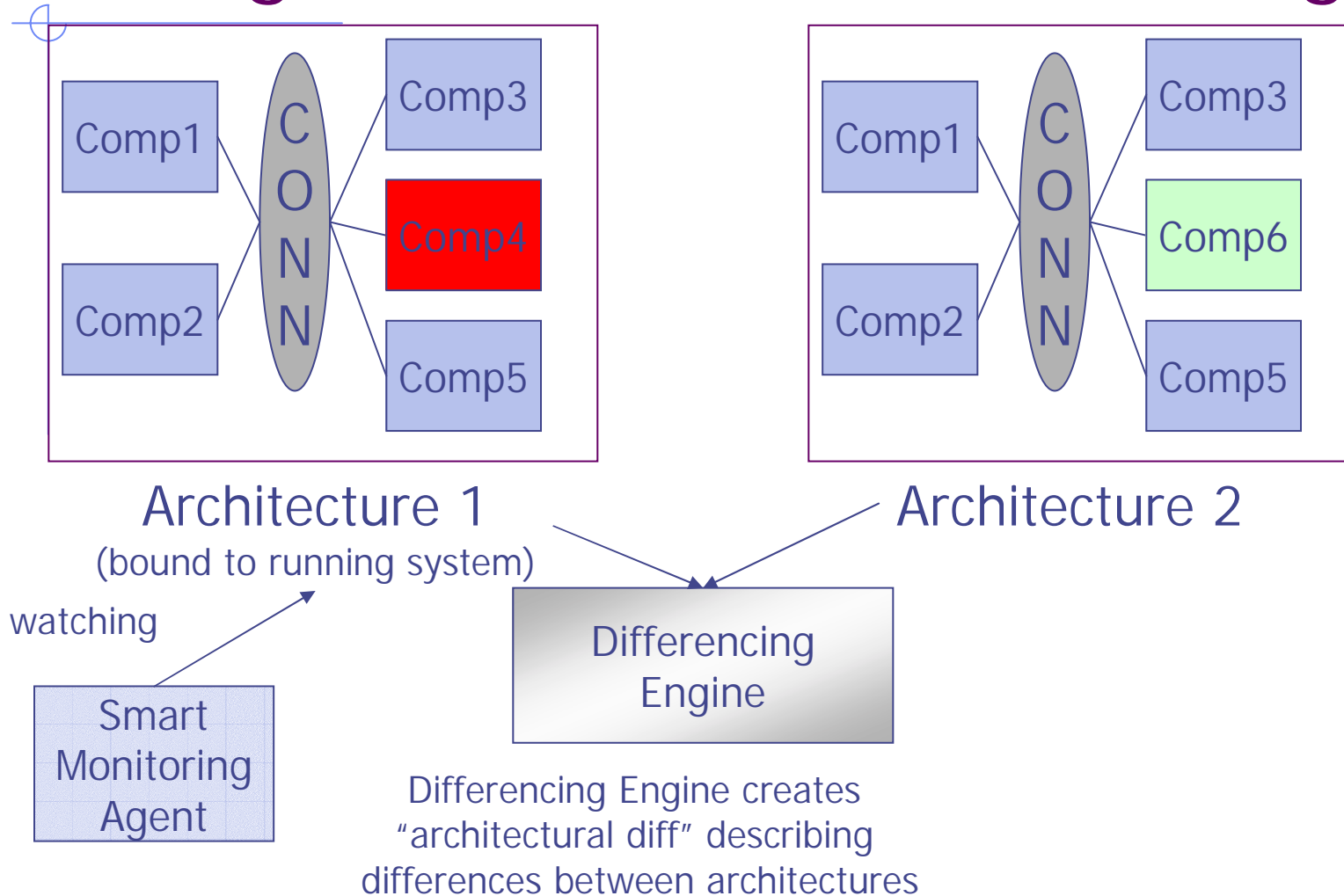


# Expressing Repair Strategies Using Architecture Differencing

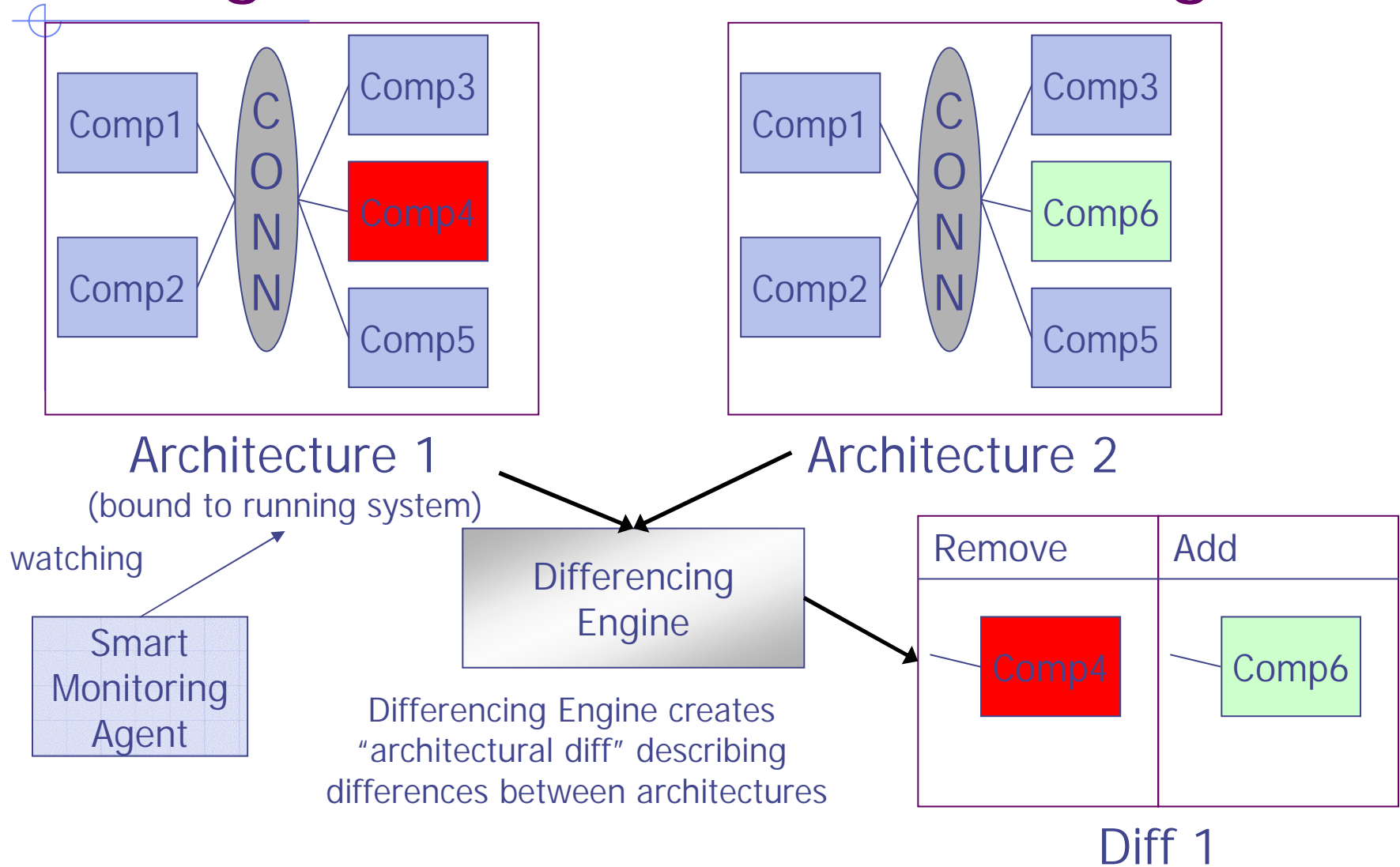




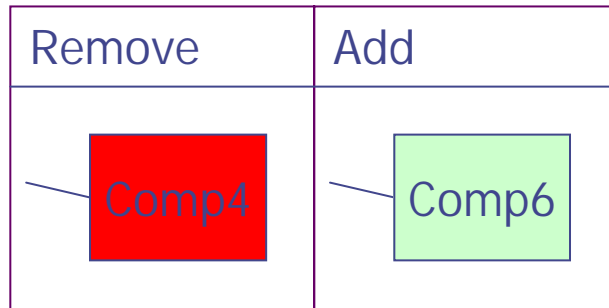
# Expressing Repair Strategies Using Architecture Differencing



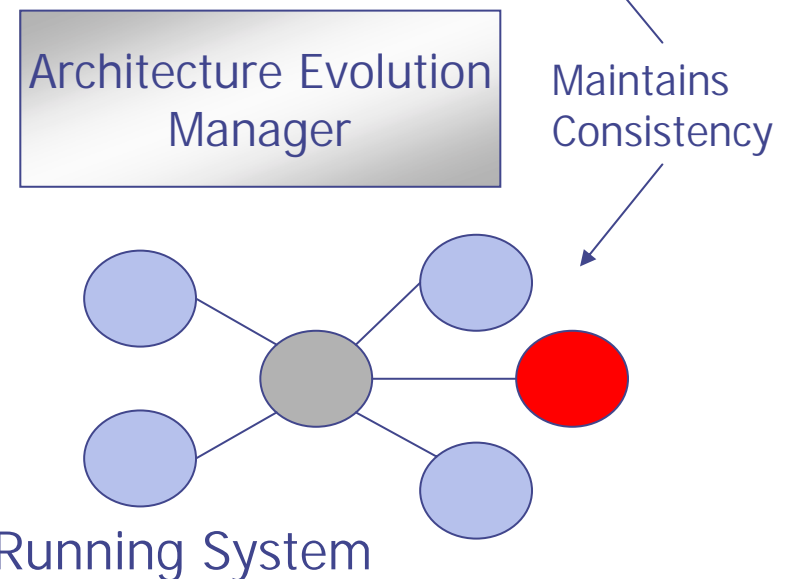
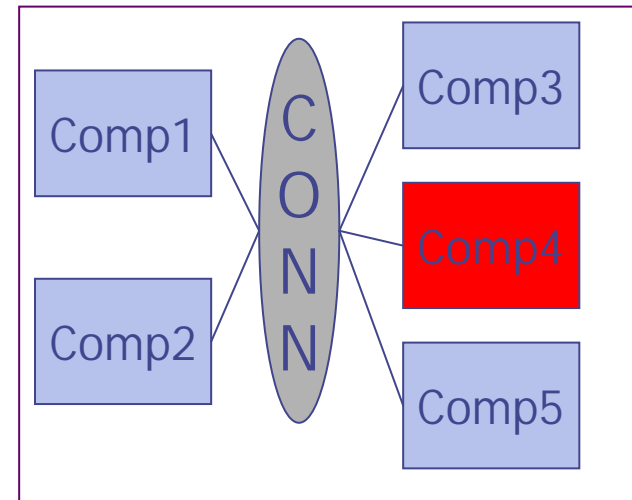
# Expressing Repair Strategies Using Architecture Differencing



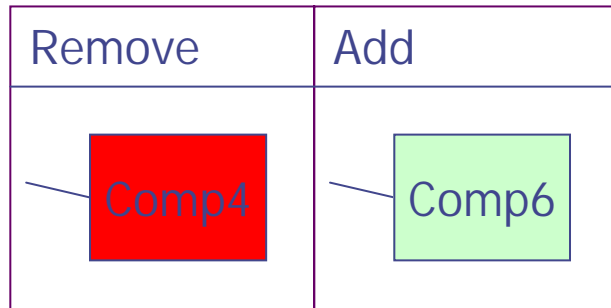
# Effecting Repairs Using Architectural Diffs



Repair Plan 1



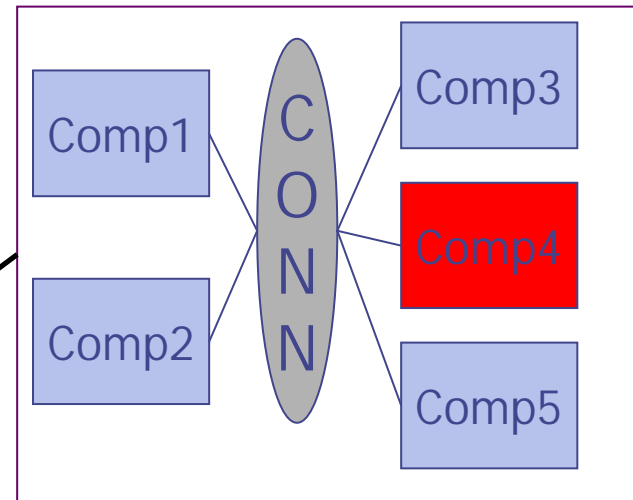
# Effecting Repairs Using Architectural Diffs



Repair Plan 1



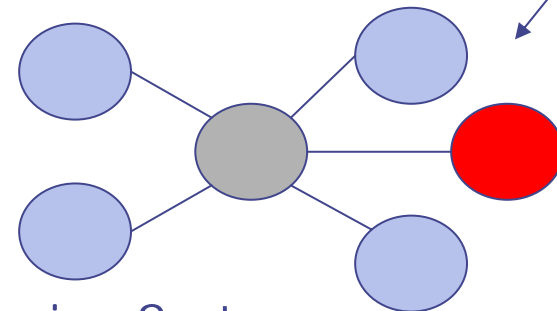
Architecture Merging engine merges architectural diffs into architecture descriptions.



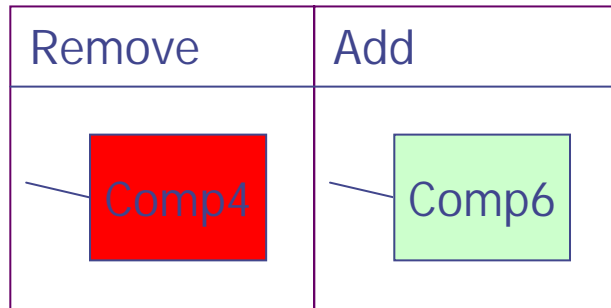
Architecture 1



Maintains Consistency



# Effecting Repairs Using Architectural Diffs

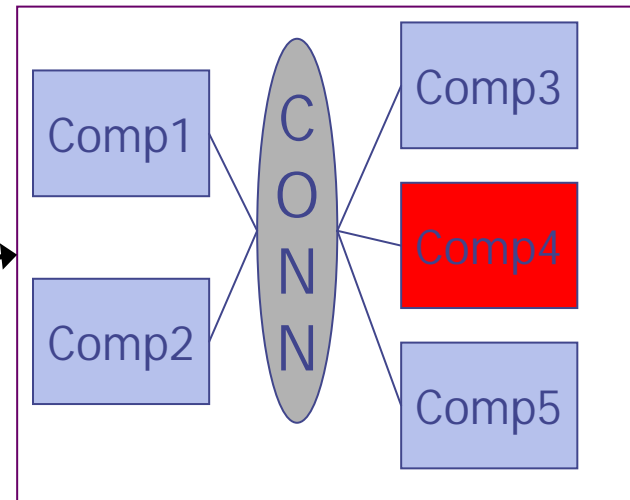


Repair Plan 1

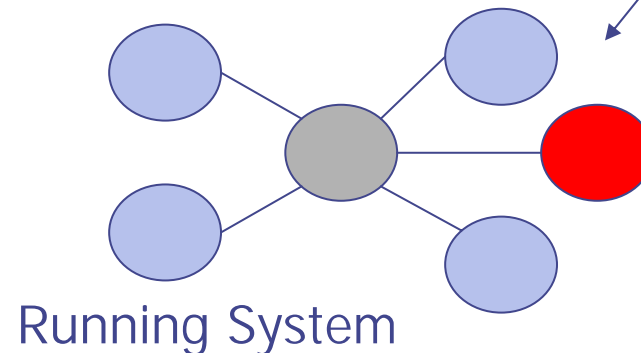


**Note:** A “what-if” merge can also be done against a copy of the architecture description for validation or analysis.

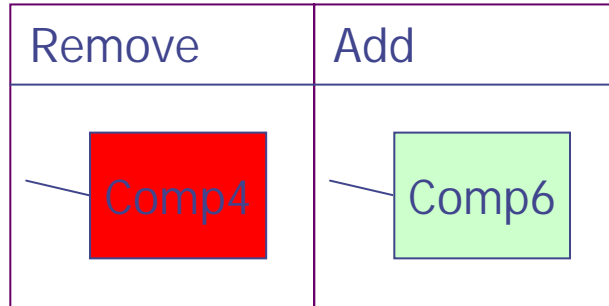
*Performs merge*



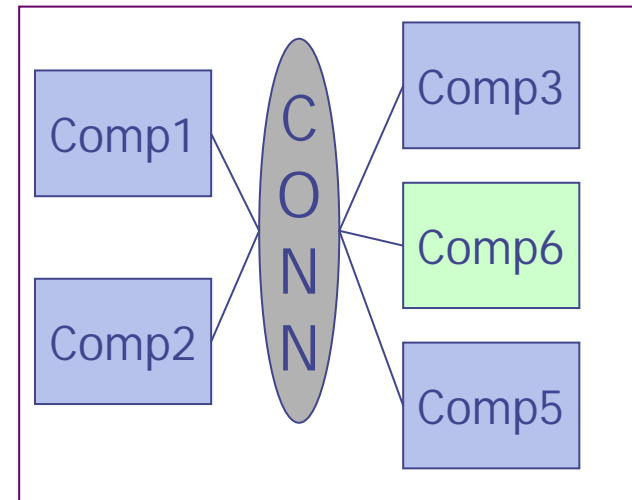
Maintains Consistency



# Effecting Repairs Using Architectural Diffs



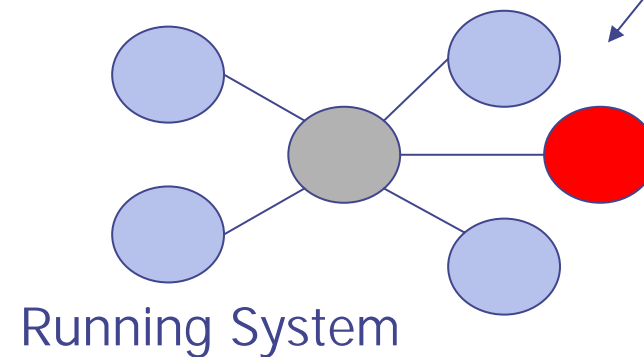
Repair Plan 1



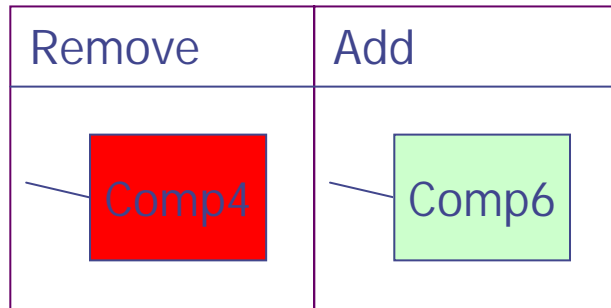
Architecture 1



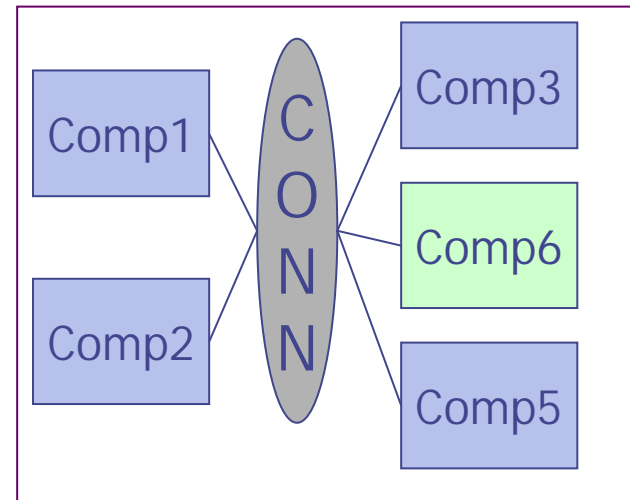
Maintains Consistency



# Effecting Repairs Using Architectural Diffs



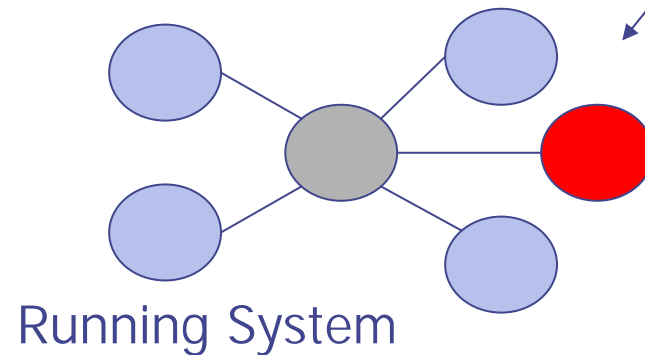
Repair Plan 1



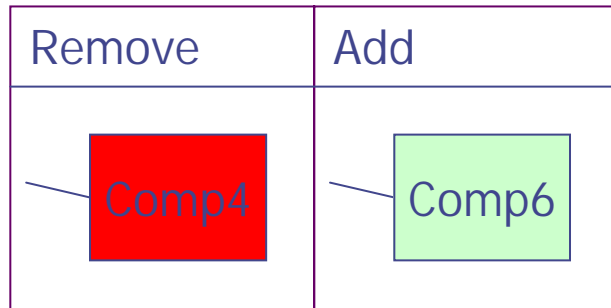
Architecture 1



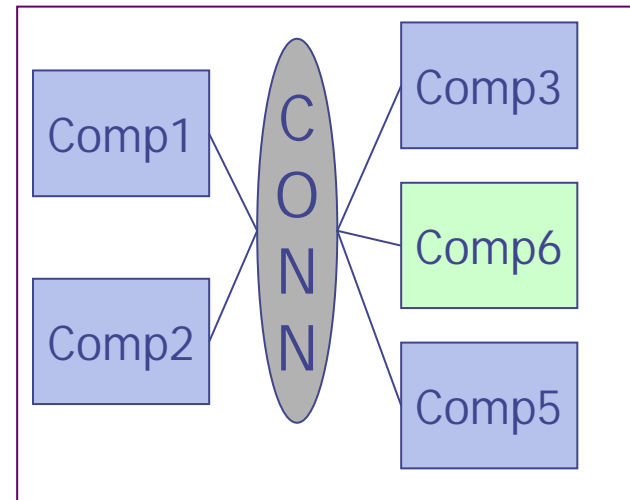
Maintains Consistency



# Effecting Repairs Using Architectural Diffs



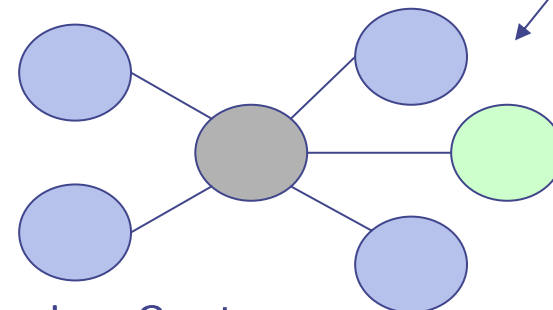
Repair Plan 1



Architecture 1



Maintains Consistency





# Applications Targeted

- ◆ Spacecraft/Spacecraft Ground Systems
  - n Architecture modeling formalism, ideas about dynamism already being adopted by MDS project at JPL
- ◆ Other component-based, event-driven systems
  - n Military command and control
- ◆ Multi-agency systems
  - n Coalition warfare among allied partners with independently developed systems

# Future Work/Top Ideas

## ◆ Distributed Dynamism

- n Making repairs in the face of
  - w (Partial) link failure,
  - w (Partial) node failure
  - w Asymmetric connectivity

## ◆ Are diffs sufficient as repair plans?

- n Ordering of changes
- n Additional information needed to make changes

## ◆ Approaches to quiescence

- n Inspired by Kramer & Magee