



# Distributed Autonomic Network Technology

Alan Clark  
CEO, Telchemy

[alan.d.clark@telchemy.com](mailto:alan.d.clark@telchemy.com)

# Outline

---

- Autonomic (and other bio-inspired) Systems
  - Why do we need them?
  - What are they?
- Autonomic Networks – some ideas
  - ANS Models
  - Some practical problems with ANS
- Distributed ANS

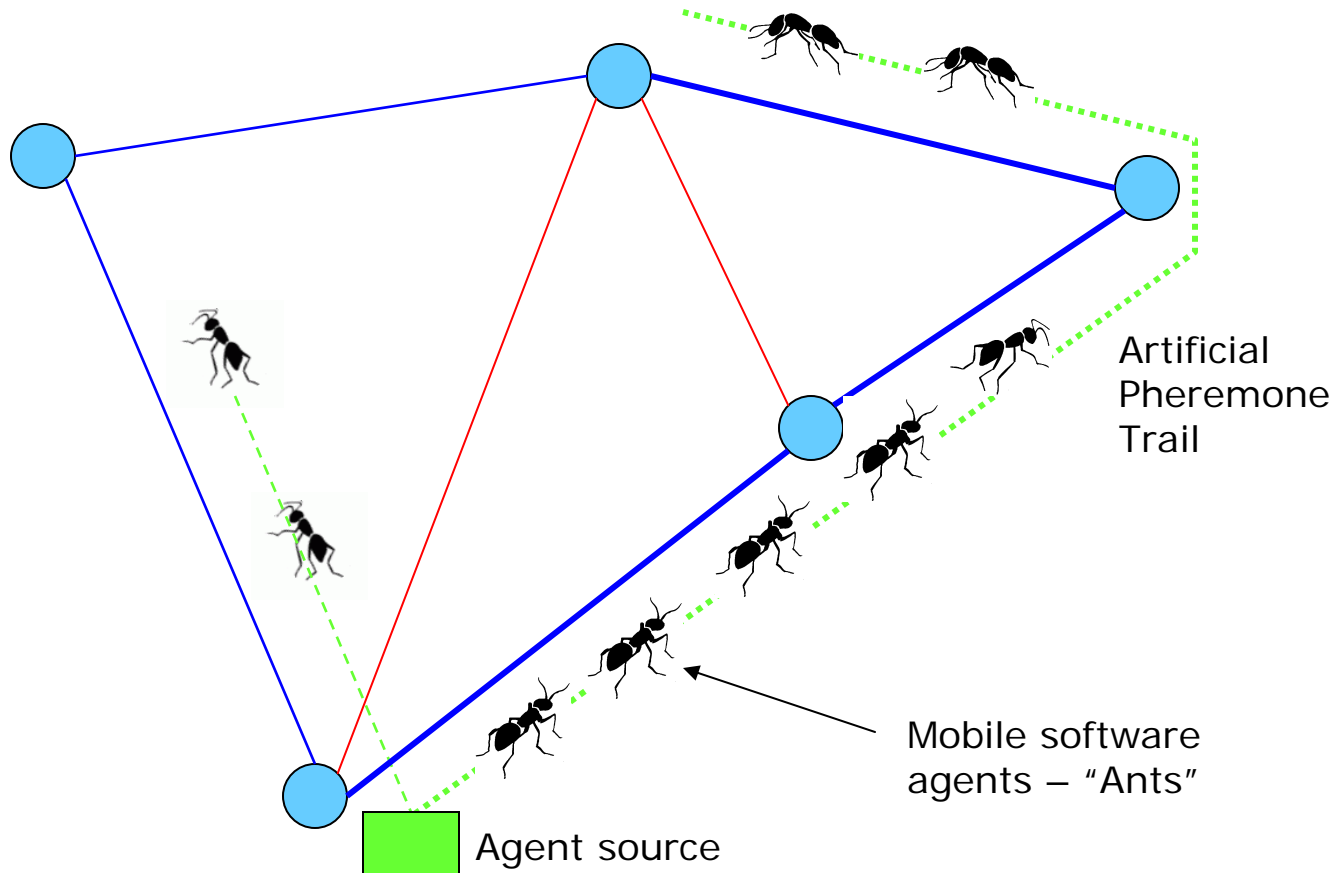
# Autonomic Systems

---

- Systems that automatically adapt to the environment and to changing conditions – patterned on the Human Autonomic Nervous System
- Similar “bio-inspired” concepts
  - 1937 - General Systems Theory (Bertalanffy)
  - 1947 – Self-organizing systems (Ashby)
  - 1948 – Neural Networks (Turing)
  - 1948 – Cybernetics (Wiener)
  - 1960’s – Evolutionary/ Genetic algorithms (Holland)
  - 1989 – Swarm intelligence (Beni, Wang)

# Example of a bio-inspired system

Load balancing in a telecommunications network



Source - Appleby & Steward (BT) 1994

(c) Telchemy Incorporated 2008

## Why do we need more “automation”

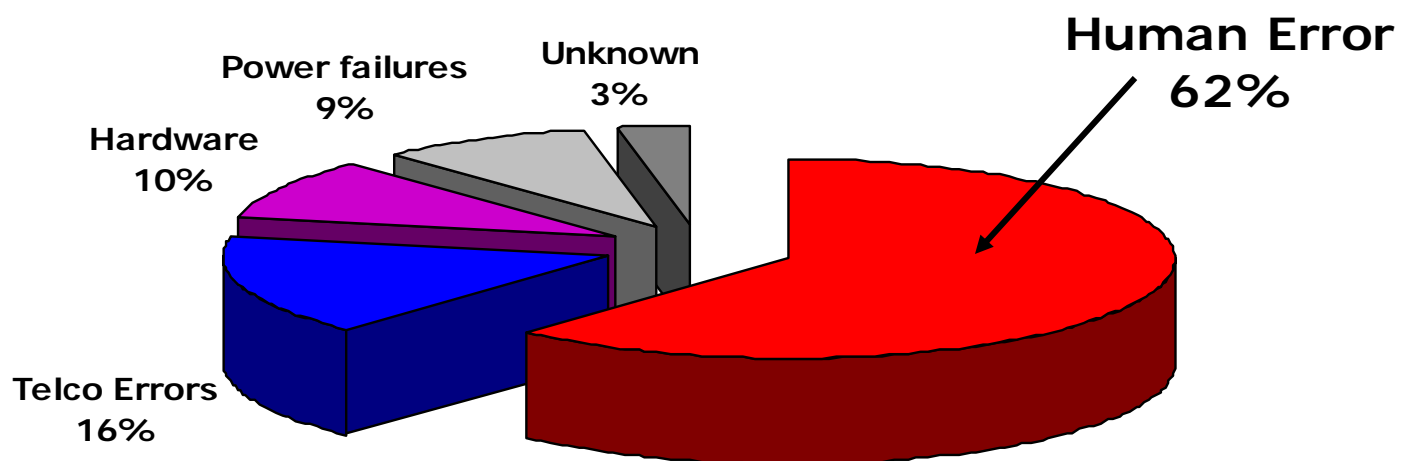
---

- Reduce network/ application downtime
  - Eliminate human error
  - Instantaneous response to fault or security problem
- Reduce capex and operating costs
  - Deliver required application performance with minimum network/ computing resource
  - Provide resilience without over-provisioning
  - Minimize “Move/Add/Change” costs
  - Ensure efficient use of “people” resources

# Why do we need more “automation”

---

Cause of Network Downtime in Multivendor Enterprise Networks



Source: Yankee Group – Network Downtime Survey 2002

# And why should this be “bio-inspired”?

---

- Wrong reasons
  - “Marketing hype” should not drive R&D
  - “Bio” does not mean “Better”
- Right reasons
  - Should be open to ideas from other fields in search for best solution
  - If it works for something as complex as the human body then it may work for networks and computing
- Right approach
  - Learn from the concepts but don’t be too literal in the implementation

# Autonomic Nervous System

---

- Central Nervous System
  - Brain and Spinal Cord
- Peripheral Nervous System
  - **Autonomic Nervous System**
    - *Sympathetic Nervous System*
      - “Accelerator”, fight-or-flight response
      - Diverts blood to muscles, increases heart rate...
    - *Parasympathetic Nervous System*
      - “Brake”
      - Diverts blood to GI tract, slows heart rate
    - *Enteric Nervous System*
      - Controls Gastrointestinal system
  - Somatic Nervous System
    - Voluntary control, stimuli (touch, hearing, sight)

# Useful concepts?

---

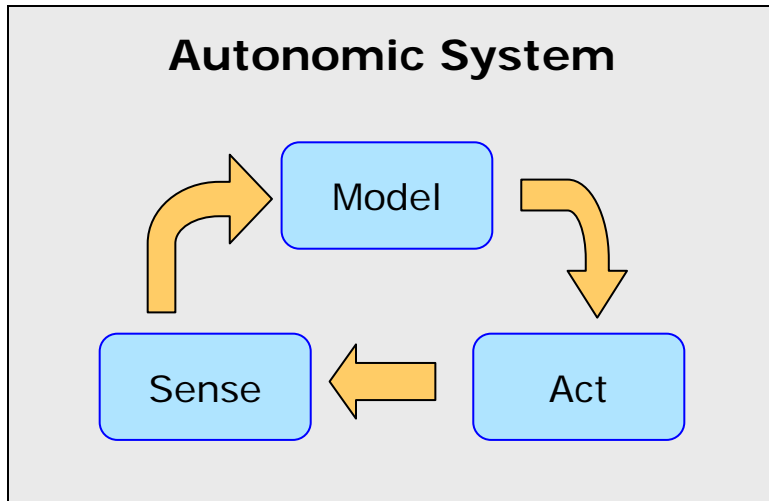
- Parasympathetic ANS (P-ANS)
  - “Normal” state – optimized for typical operating conditions
- Sympathetic ANS (S-ANS)
  - “Abnormal” state – fast reaction to failure or threat
- Interaction between P-ANS and S-ANS
  - Sympathetic ANS overrides Parasympathetic to provide short term threat protection and circumvent problems
  - Strong tendency to move back to Parasympathetic state as soon as possible

# ANS Models

---

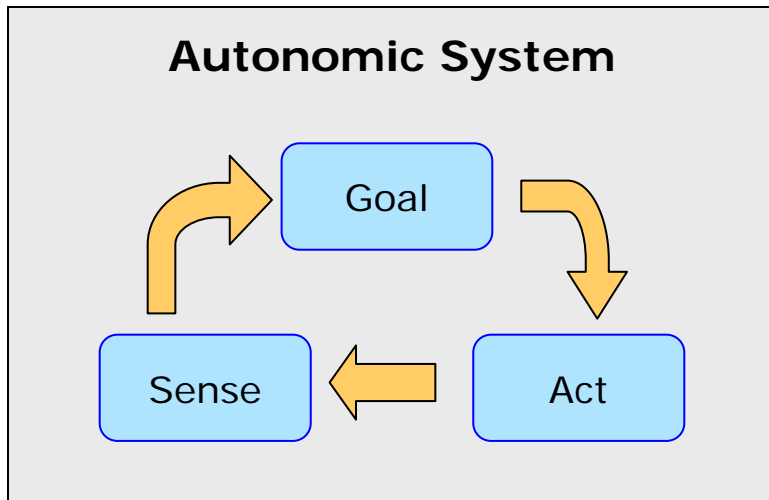
- Sense-Model-Act
- Sense-Goal-Act
- IBM's Autonomic Computing Model (ACM)
  - Monitor-Analyze-Plan-Execute (MAPE)
- IBM's Self- \* Properties

# ANS Models



## **Sense-Model-Act**

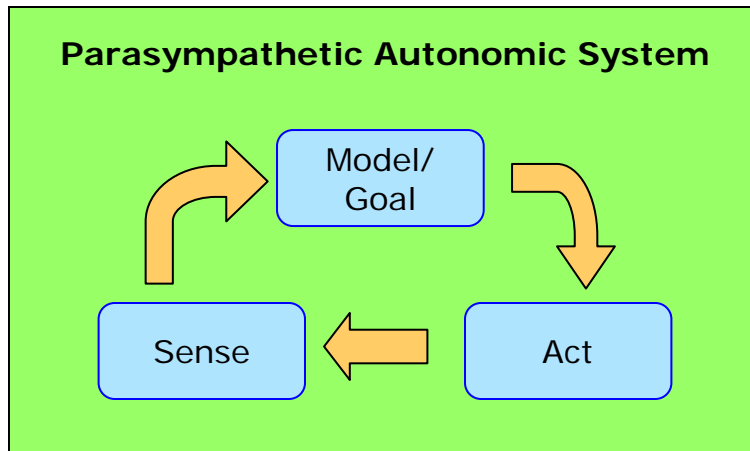
Act to bring environment toward model



## **Sense-Goal-Act**

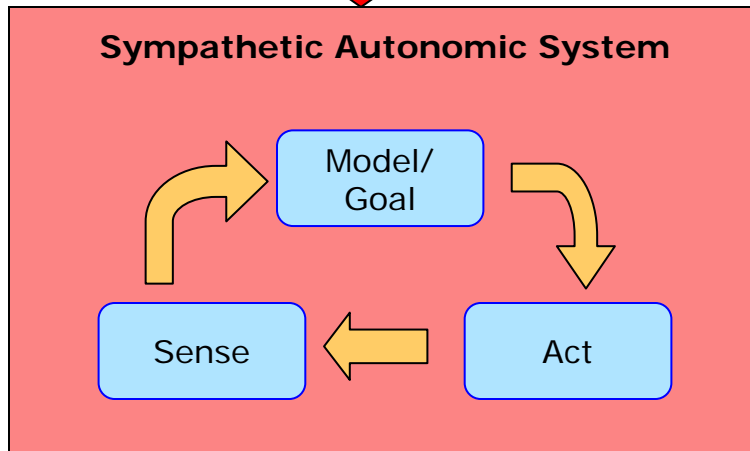
Act to achieve Goal

# Extend this model using P-ANS/S-ANS concept?



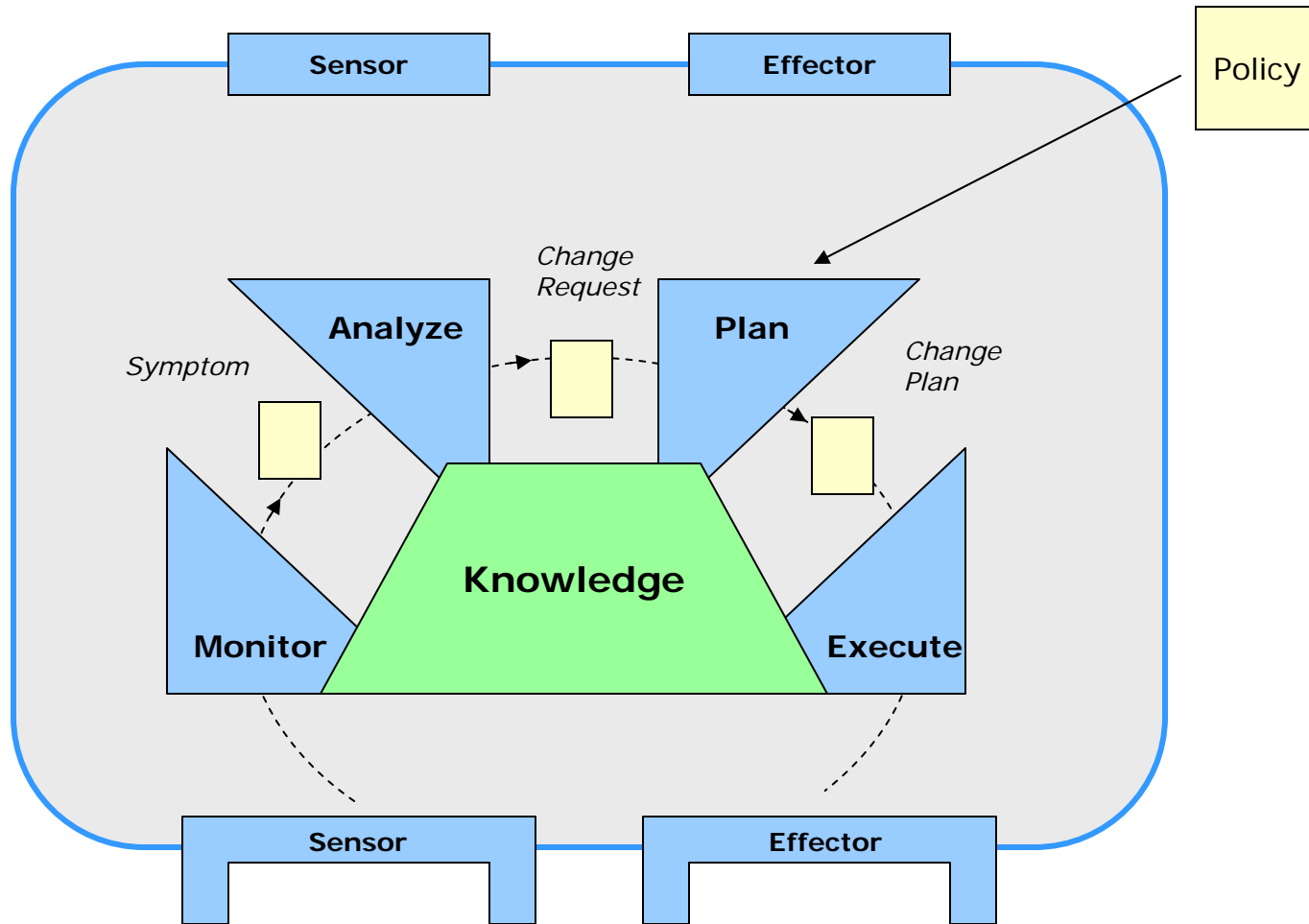
**Parasympathetic-ANS**  
Optimize performance under normal operating conditions  
- *proactive*

*Opposing models*



**Sympathetic-ANS**  
Unexpected threats, faults, sudden changes in demand  
- *reactive*

# IBM's Autonomic Computing Model



Source IBM

# Self - \* properties of an Autonomic System

---

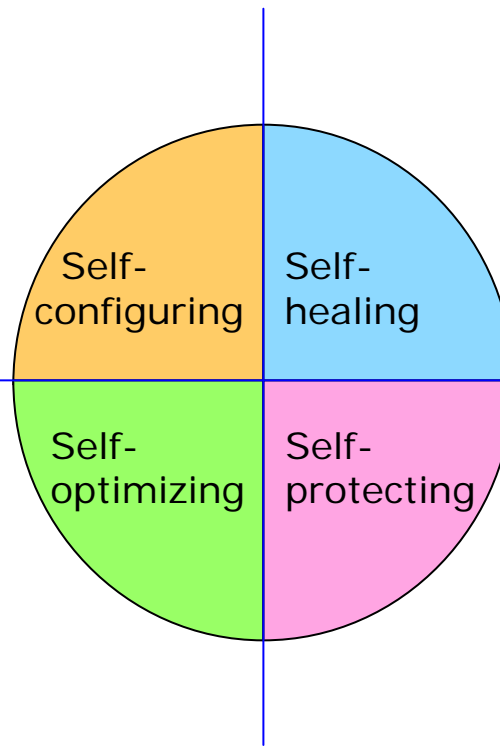
## Parasympathetic

*Increased  
Responsiveness*

*Adapt to dynamically  
changing  
environments*

*Operational  
Efficiency*

*Tune resources and  
Balance workloads to  
Maximize use of IT  
resources*



## Sympathetic

*Business resiliency*

*Discover, diagnose  
And act to prevent  
disruptions*

*Secure information  
And resources*

*Anticipate, detect,  
Identify and protect  
Against attacks*

Source (mostly) IBM

# Issues to overcome with ANS

---

- The “People” factor
  - Need to let the network administrator set the rules
  - What if systems don’t act in the way we expect\*?
  - If things go wrong and people need to get involved - how can they fix a “self-configured” system\*?
  - How to get the customer to trust their networks to autonomic systems? How to prove that performance was actually improved?
- Avoiding Catastrophe
  - Self-healing can mask gradual degradation, leading to eventual catastrophic failure\*
  - Avoiding (unwanted) emergent behavior
- Multivendor networks
  - 3<sup>rd</sup> party network devices may not cooperate
  - 3<sup>rd</sup> party network devices may have incompatible ANS-like technology and cause conflict

\* “Dealing with Ghosts”, Russell & Maglio (IBM)

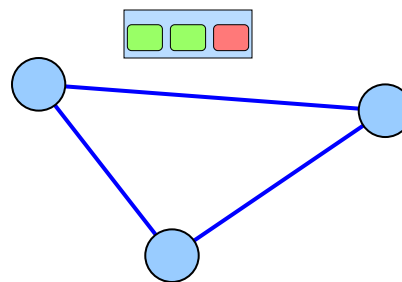
# Why *Distributed* Autonomic Systems?

---

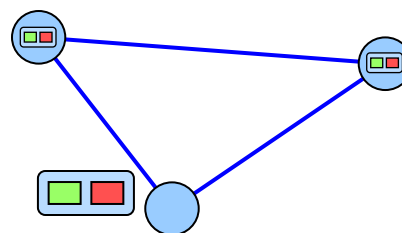
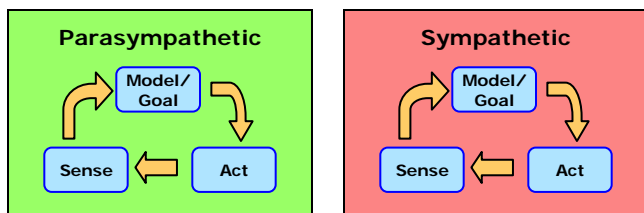
- Highly scalable
  - Processing power available increases linearly with network size
  - Reduced traffic to centralized management system
- Quicker response time
  - Instant correlation of data – preserves timing relationships and improves accuracy of problem identification
  - Localized decision process
- Response time has low dependency on network status
  - Significant problems throughout network could cause centralized systems to be overloaded
  - Continues operating if connections to NOC are degraded or down

# Distributed ANS Architecture - 1

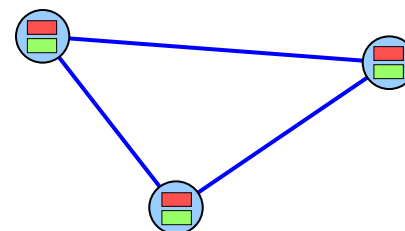
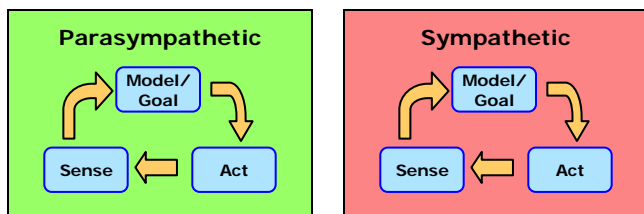
## Centralized – Human Interaction



## Distributed – Server/Node based



## Integrated (Distributed) – Node based



# Centralized – Human Interaction

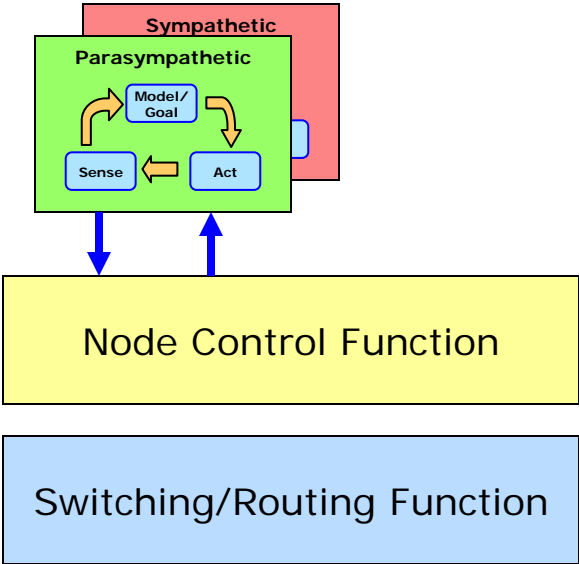
---



- **Policy** – Set by User/ Administrator
  - Constraints
  - Goals and objectives
- **Reporting** – visibility for User/Administrator
  - Network Performance
  - ANS Behavior and Performance– is the ANS detecting and correcting problems, doing better than a human in terms of speed, performance improvement, fewer iterations, less mistakes?
- **Intervention**
  - Give priority to human intervention
  - Provide immediate visibility of system state and configuration

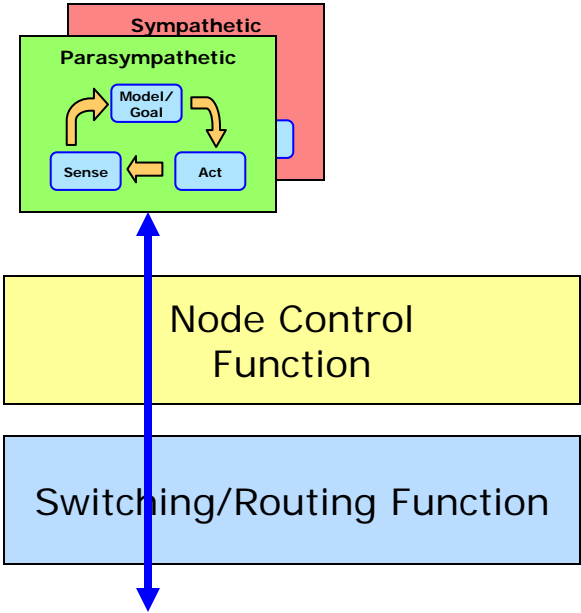
# Integrated vs Distributed

**Integrated** – interacts with local control function, “part of node”



Allows tighter integration for single vendor network

**Distributed** – resident on node but independent from it

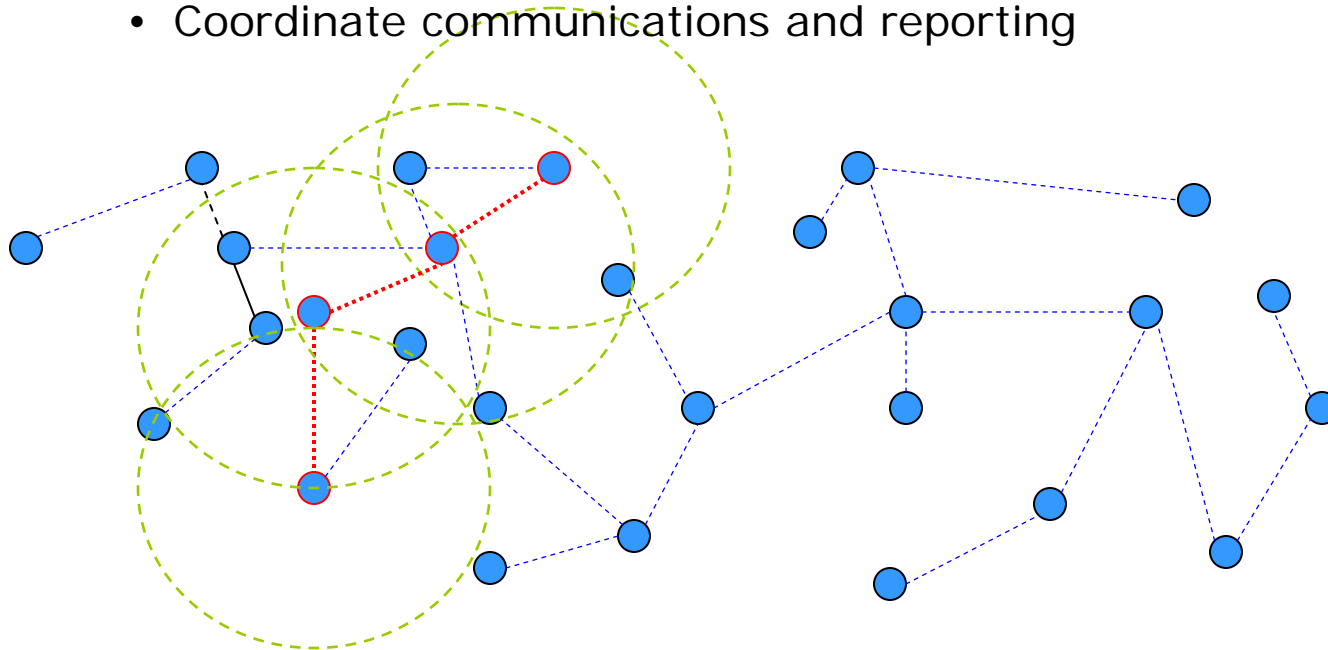


Can be used in multivendor network

# Self-organizing networks

---

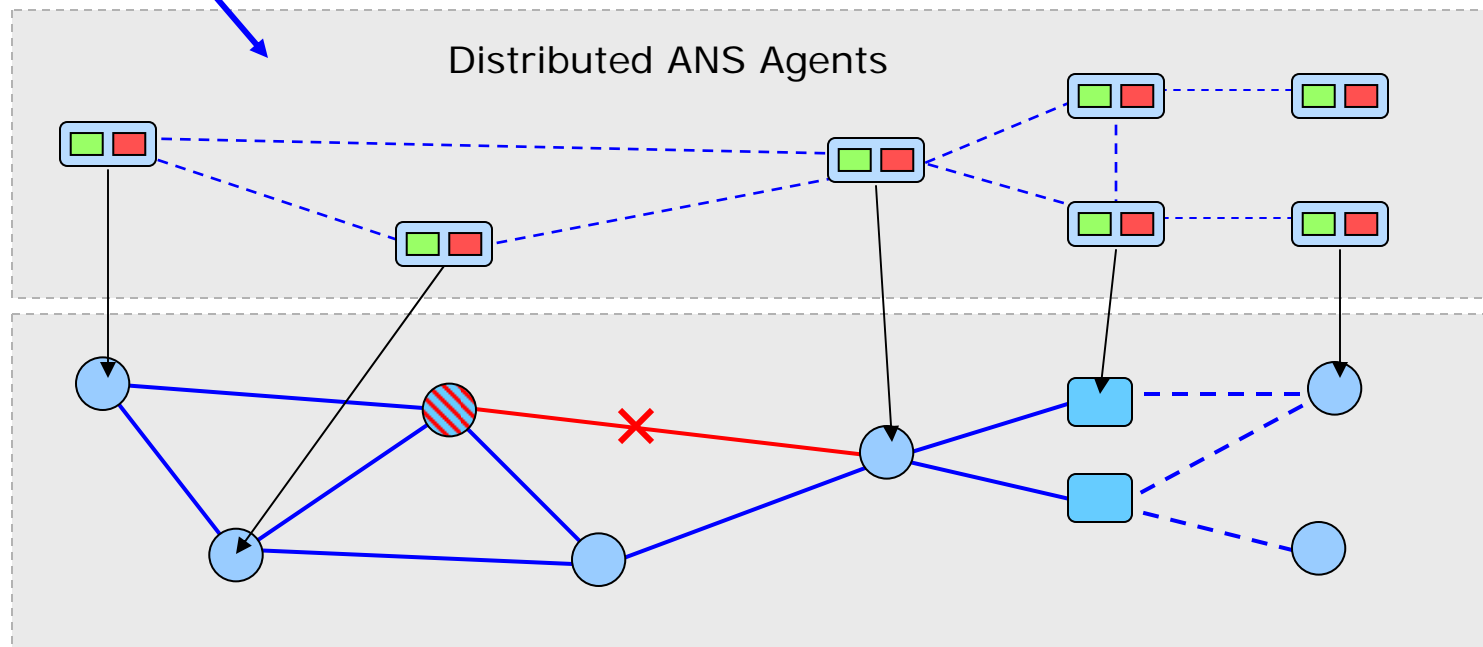
- Sensor networks
  - Networks of self organizing agents
  - No defined topology or structure
  - Sensor agents cooperate to
    - Locate and track “target”
    - Coordinate communications and reporting



E.g. Dressler, “Self-Organization in Sensor and Actor Networks”

# Self-organization amongst ANS agents

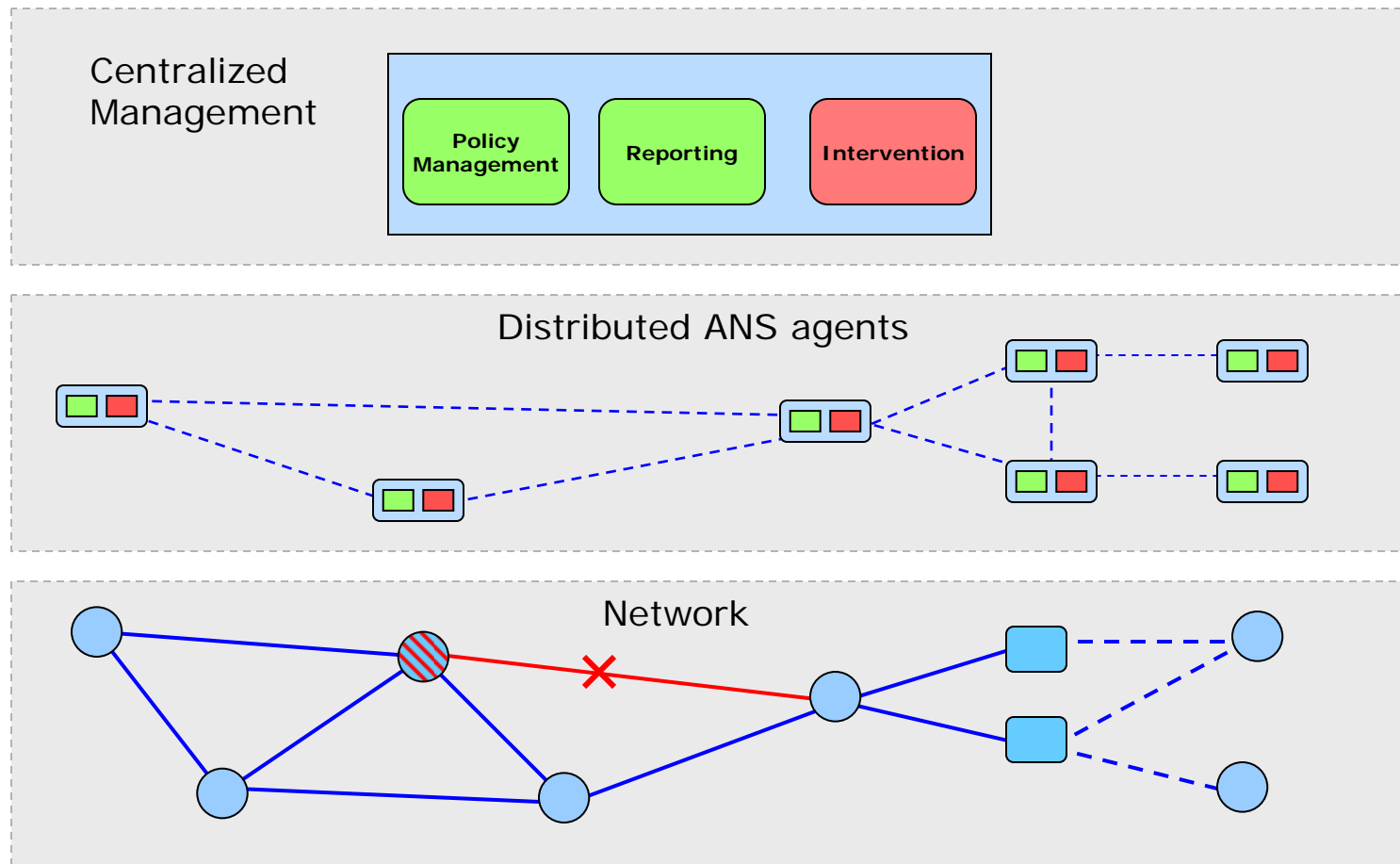
Based on Sensor Network concept



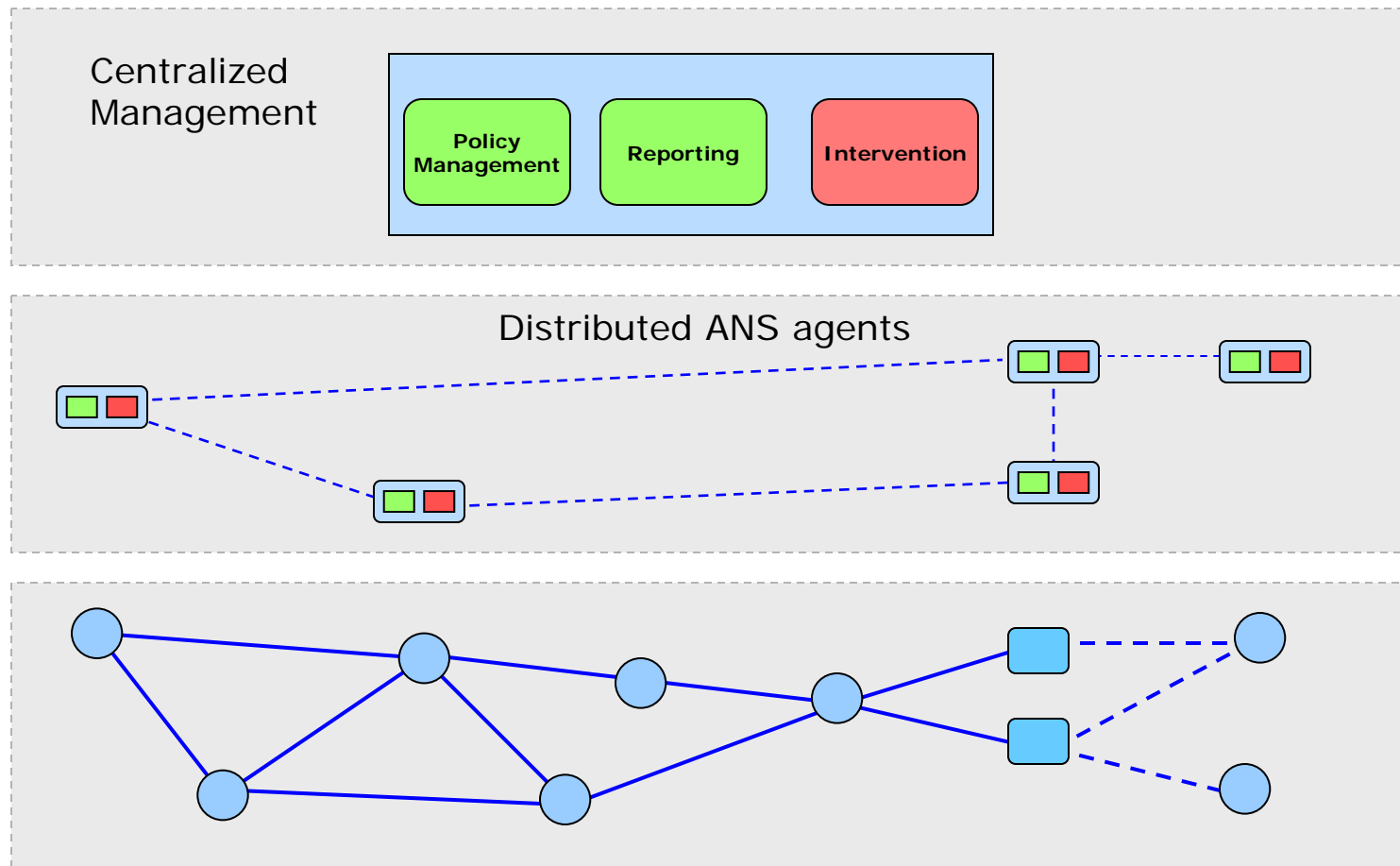
Interconnections between network Elements focus on efficient use of bandwidth

Mobile nodes, Ad hoc network

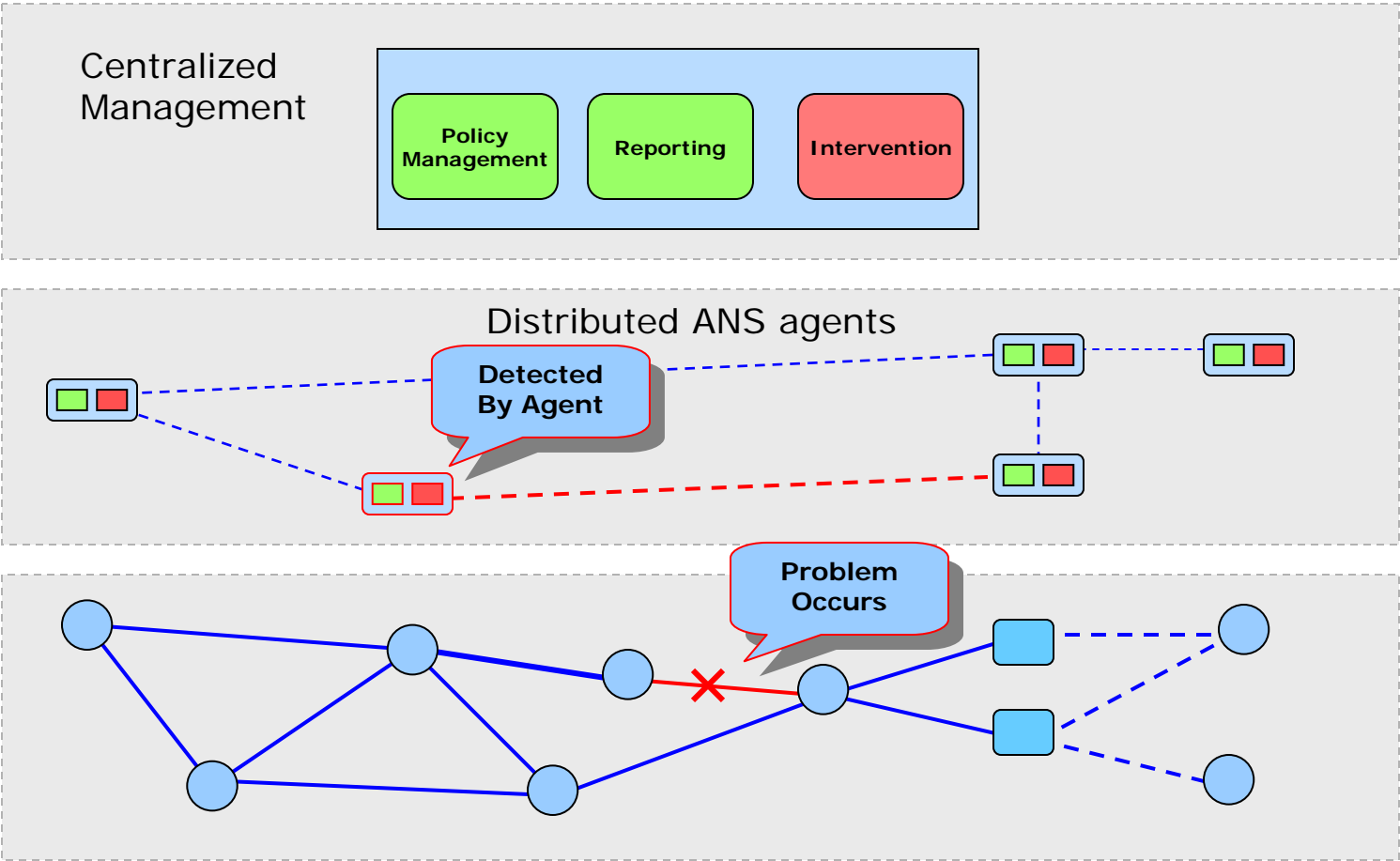
# Distributed ANS Architecture - 2



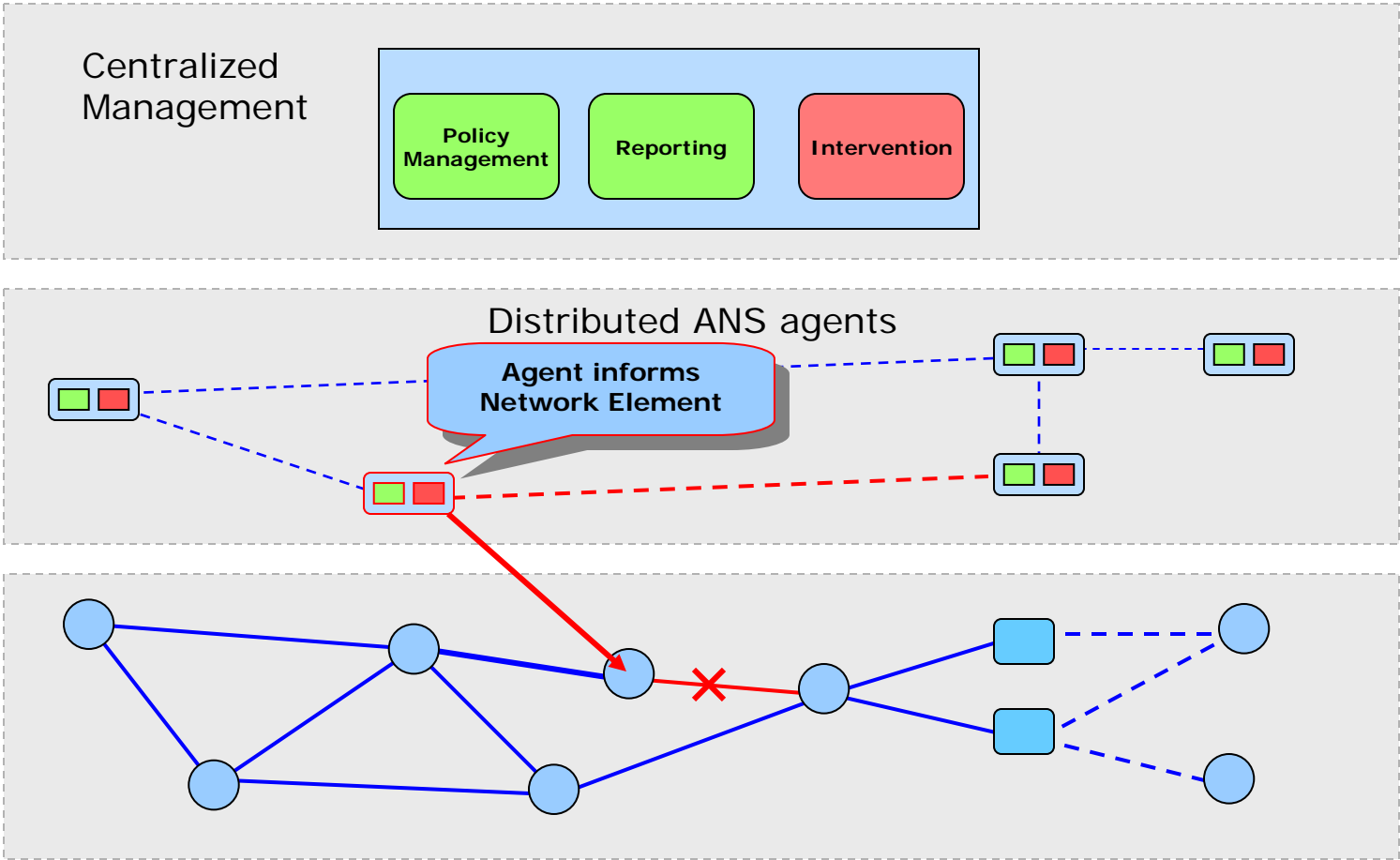
# Distributed ANS Architecture



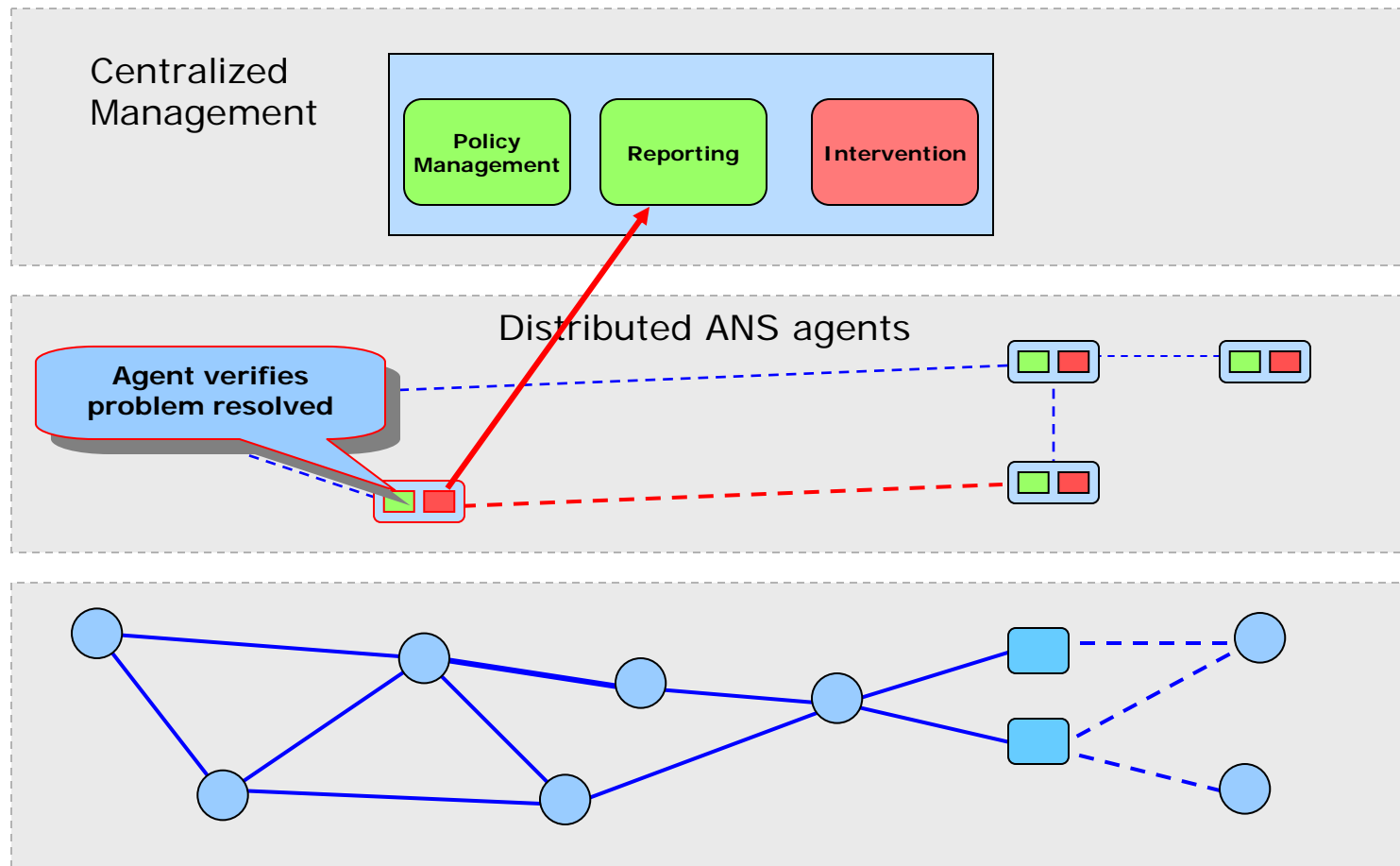
# Distributed ANS Architecture



# Distributed ANS Architecture



# Distributed ANS Architecture



# Summary

---

- Autonomic (and other bio-inspired) Systems
  - Why do we need them?
  - What are they?
- Autonomic Networks – some ideas
  - ANS Models
  - Some practical problems with ANS
- Distributed ANS