How *decentralised control techniques* can be used in building distributed systems?

- Large population of entities that independently gather information and decide how to behave;
  - entities with simple functionality and limited communication;
  - macroscopic behaviours from microscopic behaviours and interactions;
- Emergent outcome would be to provide solutions to specific problems;
There are two views:

- **distributed systems (consensus) and computational intelligent (emergence/learning);**
  - **consensus:** protocols to reach agreement under particular failure loads;
  - **emergence:** a collection of simple autonomous and self-sufficient entities that are able to adapt to changing environments;
  - **learning:** refers to the automatic mining of information from available data to create knowledge;
Self-*(management, healing) requires system support adaptation to changes:

- User needs, environment, resources, faults;
- Automatically, optimal and stable;
Distributed Systems 2

- Self-Managed Decentralised Systems using K-Components and Collaborative Reinforcement Learning
  *Jim Dowling, V. Cahill*
  - architectural reflection for reifying the internal structure of components, and contracts for reasoning about adaptation.

- An Extensible Framework for Autonomic Analysis and Improvement of Distributed Deployment Architectures
  *Sam Malek, M. Mikic-Rakic, N. Medvidovic*
  - framework for analyzing and improving distributed deployment architectures via run time redeployment.

- Network Configuration Management via Model Finding
  *Sanjai Narain*
  - theory of configuration.
Questions

◆ What does “self-managed” mean to you?
◆ What aspects of the self-management problem are you addressing?
◆ What aspects are you NOT dealing with?
◆ What domains, properties, or applications are you targeting?
◆ What are the top two/three new technical ideas/approaches that you are pursuing in this work?