# A Control Theory Perspective on Configuration Management and cfengine

Mark Burgess
Oslo University College

# For the impatient...about cfengine

- Focus version – outline
  - A little background
  - A little architecture
  - Flow of “control”
  - Discrete regulation
  - Hybrid discrete/continuous regulation
  - Summary
- There is no new work here – but some important lessons

# What is cfengine?

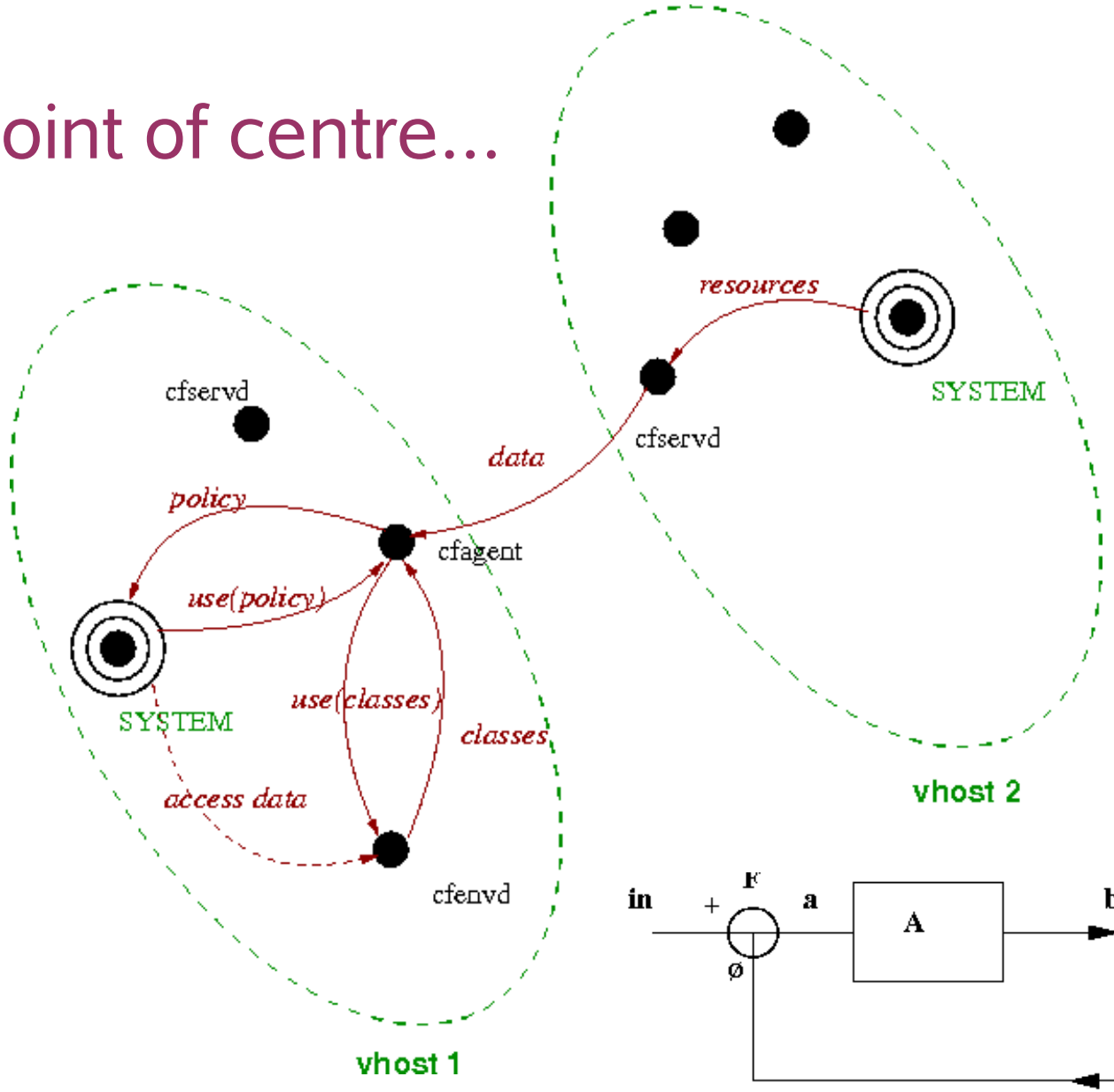
- Cfengine was born around 1993 at Oslo University and followed me to OUC in 1994.
- Currently used all over the world on estimated million machines.
  - Big and small companies, Uni's: IBM, HP, Redhat, Motorola etc. *(Even if they're not supposed to!)*
- Not a `management tool' like Tivoli/Openview
  - Automated, autonomous, distributed cooperation
- Policy based regulation engine, with “SOA”
  - Regulate resource config, on per host basis
  - Powerful abstraction model, with simple data-model

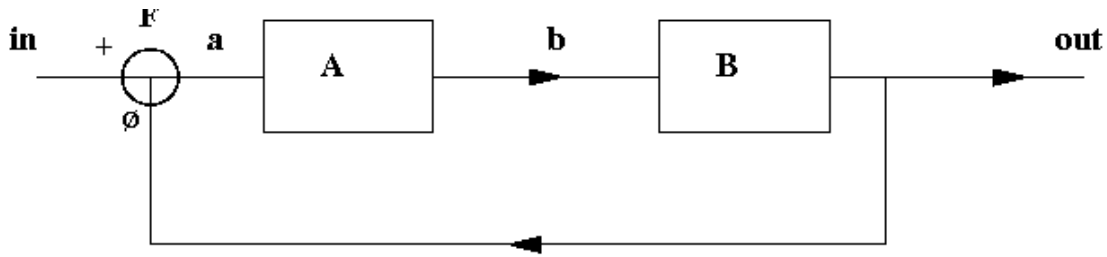
# A little architecture

- Main parts:
  - Cfagent – dangerous end, doing part
  - Cfservd – file transfer service
  - Cfenvd – resource anomaly detection service
- Principles of control
  - Cfagent read policy input and correct mismatch
  - No one can force cfagent to accept new policy (autonomy)
  - Policy decides what is “acceptable” and what is “preferred”
  - Cfagent(state) -> policy_state

# Flow of “control”

No point of centre...



# 1. Configuration regulation

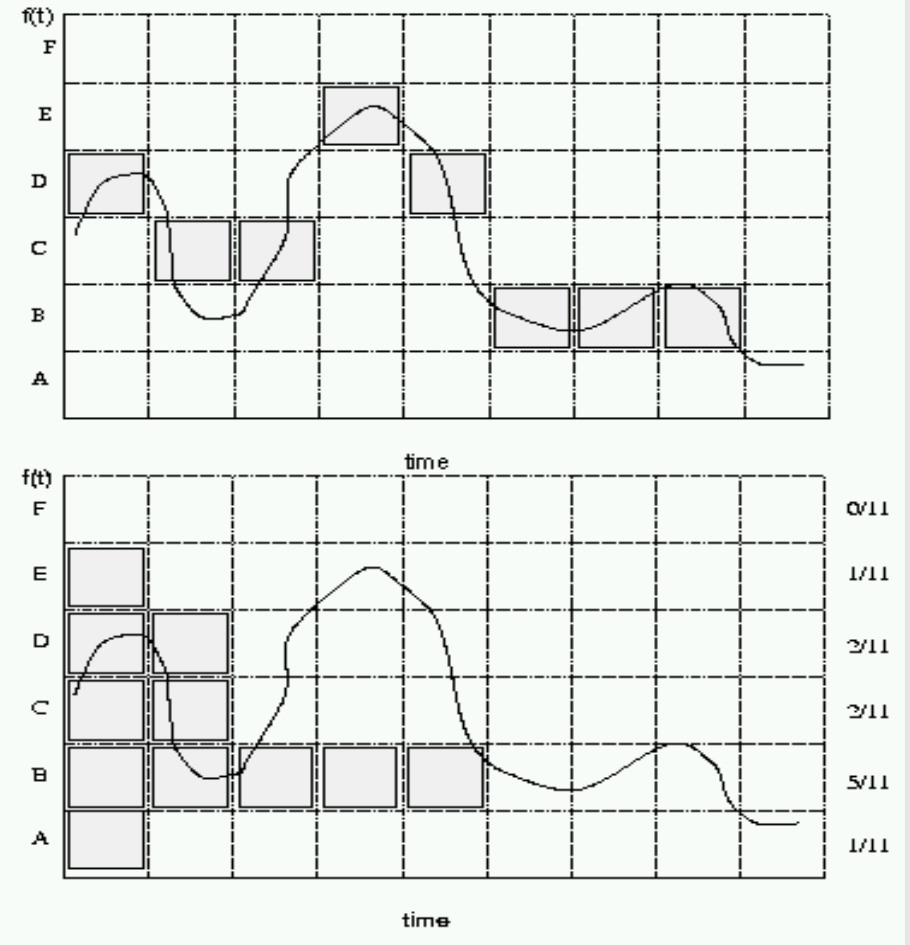
- **Configuration is <u>now</u> and is <u>discrete</u>**
- Let *p* be an operator in an alphabet {A,B,C ..} formed from {Properties × Objects} that sets state
- A policy ***P*** is a partially ordered sequence of symbols *<p1,p2,..> from this alphabet*.
- Iff we arrange for an operator p to be convergent at a point p(x) = (x), where x is policy then regulation is achieved by continually replaying ***P***.
- Fault arrivals can be handled in batch (inventory) or event driven (signalling) to within some $\Delta$t.
- *P(ABCD) -> P(ABXD) -> P(ABCD)*
- i.e. an autononic encapsulation of Shannon process

# 2. Usage regulation

- **Resouce usage is <u>learned</u> (past) and <u>uncertain</u>**
- Treat by averaging statistics, mean and deviation with periodic model (research shows weekly pattern)

- **MO**: Measure, compare, classify, inform agent, <u>conditional policy</u> for discrete follow-up.

- Information extraction all as discrete convergent operations
- Again, uncertainty like Shannon capacity theorem



# Summary

- Widely used tool for regulating policy at low level
- Simple concepts, powerful results
- Autonomy and decentralization are central doctrines :-)?!
- Events or batch – *all about Δt, nothing more*
- How does this relate to control theory?
  - Judge for yourselves

http://www.cfengine.org