Policies are the new Features

- Communications is getting personal
  - always on, context ...
- Users must be able to define their own policies
  - maximal customisability
- Questions
  - how can user’s define policies?
    - language, supporting architecture
  - what should users be allowed to do?
    - security & reliability vs. controllability & customisability
  - Charging
    - ?

Policy Conflict & FI

- Potential for conflict is greater than with FI
  - end users are usually lay users
- Can a policy language remove potential?
  - No
  - ... at least not while maintaining expressibility
- Conflict handling must be supporting architecture:
  - deployment: static analysis and suggestions
  - enforcement: dynamic analysis and resolution

Policy Language Requirements

- The Language
  - several “layers” reflecting hierarchy of users
    - natural language, technical level, ... all mapped into most basic for deployment/enforcement
  - common core + domain specific extensions
    - actions, triggers, environment variables
  - priorities
    - allow for user to fine tune their expectations
    - help with runtime resolution
  - user centric view
    - not primarily about access to and control of resources
- The supporting Architecture
  - deployment and enforcement support, conflict handling