Policy Issues for Pervasive Computing

Morris Sloman
Imperial College London,
Department of Computing
m.sloman@doc.ic.ac.uk
www.doc.ic.ac.uk/~mss
Contents

- What is pervasive computing
- Healthcare scenario
- Policy-based autonomic management
- Conflict Analysis
- Pervasive Computing Grand Challenges
- Conclusions
What is Pervasive Computing?

Technology View
- Pervasive, Ubiquitous, Sentient
- Intelligent devices everywhere: home, office, street, car, trains, on-body, implanted, in appliances – 100K devices per person?
- Mobility of people, environment, and programs
- Battery powered
- Capable of wireless communication

User View
- Devices mostly invisible – devices interact implicitly with each other and environment
- Augment human abilities in performance of tasks

The UbiComp Challenge!

- How to build the truly invisible intelligent environment?
- Designed rather than ad-hoc implementation
- Understandable
- Analysable based on underlying theory
- Manageable
- Dependable and secure
- Does not infringe privacy

Need both design and science
Healthcare Everywhere

Applications
- Automated monitoring
  - Implanted devices
  - Smart clothing
  - Swallow/inject intelligent sensors and actuators
  - Reaction to complex drug regimes
- Health advisor

Benefits
- High lower risk monitoring
- Mobility for chronically ill
- Greater out-of-hospital patient management
- Mass data & analysis
- Emergency feedback or response

Wireless video camera pill
Cardiac Monitoring

Primary unit
- RF
- Control
- Power

Secondary unit
- Antenna
- RF
- Battery
- Control

Signal conditioning

Tertiary unit/Central Server

UbiMon
www.ubimon.net
Body Sensor Nodes

- **TinyOS**
- **TI MSP430 ultra low power processor**
  - 16 bits RISC processor
  - 64KB + 256B Flash memory
  - 12-bit ADC
  - Very low power
- **Chipcon CC2420 RF module**
  - IEEE 802.15.4 (Zigbee) wireless link
  - 2.4GHz, 250kbps
  - Low current consumption (RX: 19.7mA TX: 17.4mA)
  - Hardware MAC encryption
  - Range 50m
- 6 analog channels (connect up to 6 sensors)
- 512kByte serial memory
Context Awareness

- **Context defined by:**
  - **Current location**
    - Need location detection eg GPS or base station
    - Indoors – sonic or ultrawideband wireless tags \(\rightarrow 10\text{cm}\)
  - **User activity**
    - Walking, driving a car, running for a bus – how to detect this?
  - **Ambient environment**
    - In theatre, alone, in meeting
  - **Device capabilities**
    - Screen, input, processing power, battery life ….
  - **Current QoS availability** – particularly for radio links
  - **Fusion of information from multiple sources**
Security Wish List

- Lessons from history: **everything worth hacking gets hacked**
- SECURITY solutions that are proactive, minimally intrusive, easy to use
- Need for secure ‘out of the box’ set up
- Devices that recognise/respond to “owners” only
- Means of tracing stolen devices, proving transactions
- Ability to be invisible or anonymous when needed
- Protection from spam, viruses, denial of service, identity theft etc......

**SECURITY solutions that are adaptive and context-aware**
Management – the nightmare!

- Huge, complex systems
  - Billions of processors
  - Multiple organisations
  - Managing physical world, controlling sensors, actuators
  - Humans will be in the way
- Hacker and virus paradise
- System propagates false information about individuals or organisation
- Complexity of s/w installation on a workstation or server – how do you cope with billions?
- Cater for huge systems + scale down to body area networks
**Autonomic Management**

- **Autonomic** – self-organising, self-configuring, self-healing, self-optimising, adaptive management
- Remove human from the loop
- Intelligent agents, mobile agents, policy, genetic algorithms?
Dear World,

I just can’t take it anymore! I’ve decided.
Self-Managed Cell (SMC)

- Policy
  - Service Goals
  - System Policy
  - Domains
- Context
  - Current & Past
  - History
- Cell Component Discovery
- Analysis, Simulation, Optimization
- Security etc.

Event Bus, Event Correlation

"Cell Components"
Policy-Based Management

- Managed Objects
- Events
- Monitor
- Control actions
- Decisions
- Manager Agent
- Policies
A Ubiquitous Control Loop

Master Control

Home Appliance Control

PAN Control

Pervasive GC
Layered and Federated SMCs

- Layered SMCs: application / services / network
- Peer SMCs (peer devices, peer networks, SLAs...)

Pervasive GC
SMC Composition

- Enclosing SMC “programs” the nested SMCs
Body Sensor SMC ‘Roles’

- SMC defines role assignment policies
  + role interaction relationship policies
**Discovery Policies**

**oblig on** compDetect (compName, compRef, compType) ->
medicRole.assign (compName, compRef)

**when** compType = nurseT &
signed (compRef.getCertificate, nursingCouncil_PK )

**oblig on** compDetect (compName, compRef, compType) ->
temperatureRole.assign (compName, compRef)

**when** compType = tempSensorT

………..

**oblig every** mins (1) -> discoveryService.findNewMembers()

**oblig every** mins(2) -> discoveryService.pollMembers()

………..

poll all members of a role and remove any which have not responded after max tries.
Example Policies

tempSensorPolicies {
  oblig every mins (2) ->
    tempSensor.read (tempValue) ->
    tempSensor.tempEvent (tempValue)
    // trigger local event

  oblig on tempEvent (tempValue) ->
    pda.reportTemp (tempValue)
    when (tempValue > maxtemp)
}

SMCAuthPolicies {
  auth+ tempSensor -> pda.reportTemp
}

Deploying Policies

Event
added( temperature, t-Addr)

Temperature role
obligation policies

Authorisation policy for
temperature role to
access PDA

Temperature

Policy

PDA

Service Goals
System Policy
Domains

Temperature report events
Policies to Protect PDA

oblig on 3* fingerprintfail ->
  pda/policies/bodymonitor.disable  ->
  pda/policies/selfprotect.enable   ->
  timerEvents.trigger (shutdown, currentTime + 60)

pda/policies/selfprotect
oblig every minutes (5) ->
  sendSMS (07957341, “Stolen PDA”, “ownerID”, currentLocation)

oblig on shutdown ->
  pda.locked ->
  pda.switchoff

Assume this prevents pda from being rebooted or reset without an owner card.
Context aware policies

oblig on enterhome ->
pda/policies/trusted.enable ->
pda/policies/untrusted.disable

Oblig on leavehome ->
pda/policies/untrusted.enable ->
pda/policies/trusted.disable
Peer-to-Peer Interactions

- Assign peer SMC to pre-defined roles within each SMC
- Predefined policies specify obligations and authorisation for entity assigned to role
- Default entity (PDA) interprets obligation policies provided to it.
Other forms of cell interactions

- SMCs define implementable policies
- What about overall management strategy or goals?
- Requires goal refinement
  - Use refinement patterns in order to refine goals
    - Domain independent refinement patterns
    - Domain specific refinement patterns
  - Strategy defines how the goals are achieved. Note, there can be multiple strategies to achieve the same goals.
  - Derive strategy from goals and system description through abduction
- Very hard, not implementable on PDAs
- A Goal-based Approach to Policy Refinement by A Bandara et.al
  http://www.doc.ic.ac.uk/~bandara/publications.shtml
Policy Consistency

- Policies specify adaptive behaviour
- Multiple policies may apply to object within SMC
- Policies may be imposed by an external SMS – peer or parent
- Need to ensure consistency and coherence
Policy Conflicts

- **Policy Analysis** – Is the policy specification consistent (i.e., absence of conflicts)? Is it deterministic (i.e., conflicts can be resolved)? What properties does a set of policies satisfy?

- **Two categories of conflicts**
  - **Modality conflicts** – occur because the policies are inherently incompatible e.g., auth+ and auth-
  - **Semantic conflicts** – occur because the actions are incompatible or the policies violate desired properties of the system.

- **Potential modality conflicts** can be detected by looking at overlaps of subject, actions, target.
  - Does not take into account constraints
  - Can define precedence?
Conflict Resolution

- **Negative policies override**
  - Does not permit positive exceptions to negative policies.

- **Specified Priorities**
  - Hard to define priority
  - Several managers may specify inconsistent priority

- **Evaluating a ‘distance’ between a policy and the object to which it refers**
  - Refinement level – more concrete overrides?
  - Time of last update – more recent overrides?
  - Policy relevance – policies specified for roles take precedence over policies for SMC.
Semantic (Application Specific) Conflicts

- Types of conflict:
  - Separation of duty e.g., the same person is not allowed to authorise prescription and issue drugs
  - Inconsistency e.g. 2 policies for triggering temperature event with different maxtemp values
    Multiple policies performing mutually exclusive actions, triggered by same event
- Need to specify the conditions which result in conflict
- Constraints on a set of policies (Meta-Policies).
Most conflict analysis does not take into account policy constraints.

Need to analyse the “behaviour” of the system under a given set of policies.

Need to identify:
- Which sequence of events leads to a policy conflict.
- What consequences policies have when applied to the system.
- Check that under a set of policies the system satisfies desirable properties.

Requires: a behavioural model of managed objects e.g. statecharts, logic reasoning (complexity?, decidability?, etc.)
Using Event Calculus to Formalise Policy Specification and Analysis
http://www.doc.ic.ac.uk/~bandara/publications.shtml
Policy Analysis

- Specification errors
  - Empty subject/target domain.
  - Unsupported operations in action specification.
  - Unsupported attributes in constraint specification.

- Modality conflicts
  - Authorisation modality conflict (auth+ / auth-).
  - Unauthorised obligation modality conflict (oblig / auth-)

- Semantic conflicts
  - Separation of duty / interest conflict.
  - Application specific consistency conflicts.

Deductive reasoning over system organisation model and policy specification.

Abductive reasoning over system behaviour model and policy specification.
The UbiComp Challenge!

- How to build the truly invisible intelligent environment?
- Designed rather than ad-hoc implementation
- Understandable
- Analysable based on underlying theory
- Manageable
- Dependable and secure
- Does not infringe privacy

Need both design and science
Privacy

- You are now predictable
  - System can co-relate location, context and behaviour patterns
- Do you want employer, colleagues or insurance company to know you carry a medical monitor?
- Tension between authentication and anonymity – business want to authenticate you for financial transactions and to provide ‘personalized’ service
- Users should be aware of being monitored
- Ability to control who/what has access to “my” data (stored, communicated, inferred), ability to define levels of privacy, trust etc
We have theory for design and analysis of complex buildings, bridges, electronic circuits.

We need the theory to understand and model complex interactions of pervasive systems.

Currently use ad-hoc implementation, relying on skill of programmers.
Theory Challenges

- Large Scale, complex, dynamically self-modifying system, unplanned interactions …

- To develop a coherant informatic science whose concepts, calculi, theories and automated tools allow descriptive and predictive analysis of a pervasive system at many levels of abstraction

- To employ these theories to derive all ubiquitous systems and software, including languages;

- To validate all constructions by these theories and tools.
Theoretical Foundations

- **Basic notions** Automata; Relational databases; Program logics; Verification; Mathematical semantics; Type theories; . . .

- **Concurrent systems** Petri nets; Process calculi; Logics of action; . . .

- **Ubiquity** Mobility (ambients, pi calculus); Security and privacy; Boundaries, resources and trust; Distributed data; Game-theoretic models; Hybrid systems; Stochastics; Model-checking; . . .

www.cl.cam.ac.uk/users/rm135/plat.pdf
Current UK Activities

- Equator IRC  http://www.equator.ac.uk/
- DTI Next Wave Technologies  
  http://www.nextwave.org.uk/index.htm
- EPSRC WINES Program
- Mostly engineering
- Need to develop scientific theory and engineering principles in a tight experimental loop
- UK-UbiNet + Grand Challenges  
  http://www-dse.doc.ic.ac.uk/Projects/UbiNet/
Conclusion

- Currently pervasive systems are more hype than reality
- Some component technologies are available
- Technology problems - seamless communications, power
- Management problems - adaptive self management
- Security and privacy are major issues.
- Most research focuses on Engineering aspects
- No theory to underpin understanding, analysis & design
- SMC provides a scope for theoretical analysis and implementation
- Adaptive behaviour specified by policies
- Policy analysis and refinement are still difficult problems