Mobility Middleware - Naples, Italy © Kimmo Raatikainen 9.3.2006

Challenges and Solutions

Kimmo Raatikainen
Helsinki University Computer Science Department
Helsinki Institute for Information Technology
kimmo.raatikainen@{cs.helsinki.fi,hiit.fi}

My CV

• MSc: Numerical analysis
  – stiff differential equations
• PhD: Capacity planning of computer systems
  – workload modeling, response time distributions
• Post doc: Simulation theory
  – precision control in multidimensional simulation
• Professor: Leading the NODES group
  – telecom software architectures
  – real-time databases and computing for telecoms
  – mobile computing and communication
• “Next life” (under construction)
  – operating system issues in future mobile computing
Presentation Outline

• Helsinki University Computer Science Department
  – NODES Group
• Future Visions
• Results from NODES Group
Sections at Dept of CS

- Algorithms
- Information Systems
- Intelligent Systems
- Software Engineering
- **Distributed Systems and Data Communications**
  - Research in the Networking in Open Distributed Environments (NODES) Group
- Bioinformatics
- Helsinki Institute for Information Technology – HIIT
  - Join research institute of Helsinki University and Helsinki University of Technology (TKK)
NODES Group

- 2 professors
- 8 university lecturers
- c. 20 researchers in projects
  - c. 10 M.Sc. students
  - c. 10 Ph.D. students
- c. 10 Ph.D. students in industry

Research challenge:
- how and when systems can be composed of autonomous units
- how the units interact and behave as a system

Any technology distinguishable from magic is insufficiently advanced.
Gregory Benford

NODES Research Teams:
Wireless Internet, Collaborative and Interoperable Computing, Computing Architectures and Platforms

NODES Research Areas

- Wireless Internet
  - Efficient and secure always-on connectivity
  - Internet protocols in wireless environments
  - Mobility middleware
- Collaborative and Interoperable Computing
  - Inter-enterprise collaboration
  - Interoperability middleware
  - Trust management
- Computing Architectures and Platforms
  - Resource awareness and constraints
  - Secure run-time reconfiguration
  - Linux enhancements: timeliness, high-availability, compactness
Research Highlights

• TCP enhancements: RFCs, I-Ds, and Linux kernel
• IP QoS in access networks
• Mobile Middleware
  – Wireless CORBA: OMG standard
  – Wireless JAVA RMI: standardization under evaluation
  – Efficient Agent communication: FIPA standard
  – SOAP for wireless links: under construction for W3C
• WWRF Service Architecture
• Open Distributed Processing (ODP) standards by ISO/ITU
  – trading
  – type repository
  – interface references and binding

Jubilee Books

DEPARTMENT OF COMPUTER SCIENCE
SERIES OF PUBLICATIONS B
REPORT B-2004-1
The First Ten Years of
The NODES Group
Kimmo Raatikainen, editor
http://www.cs.helsinki.fi/u/kraatika/
NODES Research Impact

Open Source Software  International standards

Research

Education  Scientific Publications

Future Visions
Technology Domains

Personal Domain:

Ad-hoc Community:

Networked Services:

Digital Home Domain (and other smart places):

Other Smart Places

Office:

Shopping Center:

Private Transportation:

Public Transportation:
... but do not forget non-smart places

Objective

Solution stack as similar as possible for all technology domains
• Future applications and platforms will be context-sensitive, adaptive, and personalized.

• They need to be run, in a reasonable and secure manner, on variety of execution environments: anywhere, anyhow, anytime, by anyone

• Future systems must support self-awareness, distributability, reconfigurability, proactivity, collaboration, security, trust, privacy, mobility, diverse access technologies, extendability, incremental deployment, resource awareness, ...
**NODES Research Challenge Cube**

**Research Themes:**
- Context Sensitivity
- Security – Trust – Privacy
- Mobile Always-On Connectivity
- Interoperability

**Tools and Methods:**
- Formal Methods
- Programming Models
- Performance Analysis

**Software Artifacts:**
- Operating Systems
- Internet Protocols
- Middleware

---

**Paradigm Shift**

From technology centric to user centric
From what we have now to what we leave behind us
Six Premises

1. Forget end user terminals
   - dynamically configured end user systems
2. Stop thinking users in isolation
   - communities and buddies are important
   - users will have several roles and belong to several communities
3. Users want to be involved
   - user in the driver’s seat
4. Devices are for the users
   - understanding users
5. People are different
   - managing mass-scale personalization
6. Trust is the king
   - systems that compromise the trustworthiness abuse our right of privacy and will be considered as an insult and a fraud

Three Questions

1. Do our current operating systems support reconfigurability?
2. Are our programming models and tools adequate for context aware applications?
3. Do our current middleware solutions support development of such applications?

Answers: No, No, and No
Six Primary Research Challenges

1. Reconfigurable systems
2. Context-awareness
3. Security - Trust - Privacy
4. Software Development and Maintenance
5. Programming models
6. Wireless communications

Reconfigurable Systems

- End-user system instead of end-user device
- Research issues:
  - detection of devices
  - environment monitoring
  - event notification
  - event filtering
  - system modeling
  - configuration management
  - management of ad-hoc communities
  - group communication
  - decision rules for reconfiguration
Context awareness

- Almost any information available at the time of interaction can be seen as context information
- Research Issues:
  - extraction of context information
  - interpretation of context information
  - reasoning about the current contextual situation
  - adaptation of application behaviour
  - ways to express which pieces of information belong to the context
  - distributed (RDF/XML) data management
  - context modeling

Security - Trust - Privacy

- Security, trust and privacy must be addressed
  - from the very beginning of system design
  - on all levels: hardware, operating system, protocols, middleware
  - Trust is not of type On/Off
- Research issues:
  - protecting system against unauthorized modifications
  - program validation/verification
    - what an uploaded/downloaded piece of software really does
  - trust modeling
  - how fragments of information can be efficiently shared in a controlled manner
  - key/certificate management
  - implications of ad-hoc communities
    - what can be done without trusted servers
    - system integrity (Trusted Computing Framework)
Software Development and Maintenance

• Software Architectures
  – to get the software architecture right:
    • Not too detailed – Not too summary
  – modularity allowing exchanges
    • any block of software/hardware can be replaced
    • new “hardware” technologies can be incorporated
  – clarify thinking: architecture is a design tool
    • not to mix apples and oranges
  – currently the worst area of “NIH research”

• Software Processes
• Software Life Cycle Management
• Service Configuration and Deployment

• Targets:
  – Increase abstraction level without sacrificing performance too much
  – Increase automation (through modeling)
  – From craftwork to industry

Programming Models

• Currently Java and client/server dominates
• Many useful applications are naturally expressed according to the client-server paradigm
  – How to implement servers in proximity networks
• Context expressed as rules
  – Condition-action programming model; to specify conditions under which each action is to be launched
  – Guarded Commands: Old ideas of Dijkstra and Hoare
• Partially available systems – a new approach to fault tolerance
  – Programming models to compensate/overcome missing functionality
Wireless Communication

- air link is and will remain narrow and error-prone
- optimization on all levels is necessary
  - link, network and transport layers
  - messaging layer
  - communication languages (presentation layer)
  - interaction protocols and patterns
  - One can destroy the performance on each level!
- Quality-of-Service in wireless and mobile systems
- Group communication
- Mobility management:
  - terminal mobility
  - (sub)network mobility
  - personal mobility
  - session mobility

Conclusion

It is time to reconsider fundamentals!
Evolution is fine but you need to be ready for a revolution.

In 1972 Dijkstra defined computer science as managing complexity
Now we really need to manage the complexity
Current and Past Projects

Most of our research has been funded by EC, National Technology Agency (Tekes), and industry

Wireless Internet Project Family

- Mowgli: 1993-1999
- wCORBA: 1998-2000
- PRIME: 1998-2000
- IIP Cross: 2005-
- S4ALL: 2005-
- Fuego Core 05/7 2005-
- Dynamos 2004-
- Fuego Core 2002-2004
- Crumpet 2000-2002
- IIP Wireless 2002
- IIP Mixture 2003-2004
- IIP Mobile 2001
- TRANSAT 2001-2003
- VAAWIT 2001-2003
- SESSI 2004-2005
- MIND 2001-2002
- BRAIN 2000-2001
- ANWIRE: 2002-2004
- Fuego Core 2001
- TranSat 2001-2003
- Crumpet 2000-2002
- Dynamos 2004-
Monads

- 1998-2000
- Partners:
  - Nokia (NMP, NTC, NRC)
  - Sonera
- Topics:
  - agent communication in wireless environments
  - adaptability to available resources
  - short-term predictions of available resources

www.cs.helsinki.fi/research/monads/

Monads Communication Architecture
Monads Web Agent Architecture

Mobile Terminal

Netscape
Mowgli Web Agent
MDCS

Fixed Network

Monads Services
QoS Prediction Agent
Image Compression Agent
Monads Web Agent
Mowgli Web Proxy
MDCS
Web Server

Monads QoS Prediction Architecture

GPS
Location Service
Waypoint Map
QoS Management
Perception Service
Log
QoS Model
Route Modeler Agent
QoS Modeler Agent
QoS Prediction Agent
Client Agent
Fuego Program 2004

Charter:

to address the research challenges enabling future Internet – novel concepts, applications and key enablers

• c. 15 researchers in 2 (3) groups:
  • Mobile Computing Group
  • Prof. Kimmo Raatikainen

• Other senior researchers:
  • Dr. Ken Rimey
  • Dr. Pekka Nikander
  • Dr. Andrei Gurtov

Focus areas:

• Mobile Middleware
• Host Identity Protocol
• Personal Distributed Information

Current projects
Fuego Core, InfraHIP, Dynamos

Main Achievements

• Mobile Middleware
  – Efficient SOAP/XML messaging
    • Strong W3C involvement
  – asynchronous events and SIP interworking
  – XML-based file synchronization
  – prototype implementation for Linux and port to S-60

• Personal Distributed Information Store
  – update-anywhere replicated XML database concept
  – prototype for Linux and port to Symbian

• Host Identity Protocol
  – Linux kernel (2.6) implementation
  – IETF contributions
**Fuego Core System**

- Ubiquitous and Pervasive software
- Mobile Services
  - Presence Service
  - Sync. File System
  - Event Service
- Ubiquitous and Pervasive software
- Mobile Applications
  - Presence Client
  - Sync. File system
  - Event Bus
  - Jetty
  - Wireless SOAP
  - HTTP 1.1
  - BEEP
- Ubiquitous and Pervasive software
- Distributed Servers
  - TCP
  - Host Identity Protocol (HIP)
  - IP

**Features**

- End-to-end authentication of hosts
- Encryption of network traffic using IPsec
- Mobility and multihoming support
- Resilient sockets
- Reliable messaging
- Multiplexing multiple connections on TCP
- Prioritization of connections
- Efficient sync/async messaging
- Efficient XML serialization
- Persistent connections across mobility
- Expressive async. communication
- Content-based routing using filters
- Event buffering / mobility support
- Support for context-aware operation
- Automatic reconciliation of XML documents
- Optimized storage of XML
- Ad-hoc file sharing
- Integrates with existing file systems (NFS)
- Presence service based on events
- Control of presence attribute visibility
- Support for context-aware operation
- Presence service based on events
- Control of presence attribute visibility
Fuego Solution Stack

- Host Identity Protocol
  - End-to-end authentication of hosts
  - Encryption of network traffic using IPsec
  - Mobility and multihoming support
  - Resilient sockets
- BEEP
  - Reliable messaging
  - Multiplexing multiple connections on TCP
  - Prioritization of connections
- Wireless SOAP
  - Efficient sync/async messaging
  - Efficient XML serialization
  - Persistent connections across mobility
- Event Service
  - Expressive async. communication
  - Content-based routing using filters
  - Event buffering / mobility support
  - Support for context-aware operation
- Synchronized File Systems
  - Automatic reconciliation of XML documents
  - Optimized storage of XML
  - Ad-hoc file sharing
  - Integrates with existing file systems (NFS)
- Presence Client
  - Presence service based on events
  - Control of presence attribute visibility

Fuego Core 2005/7

- Mobile distributed events
- XML/RDF data access
- XML messaging
- Software configuration

*: external enabler
Final Message

Linux was invented here