



# A System Perspective on Cognition for Autonomic Computing and Communication

Arjan Peddemors<sup>♦</sup>, Ignas Niemegeers<sup>\*</sup>,  
Henk Eertink<sup>♦</sup>, and Johan de Heer<sup>‡</sup>

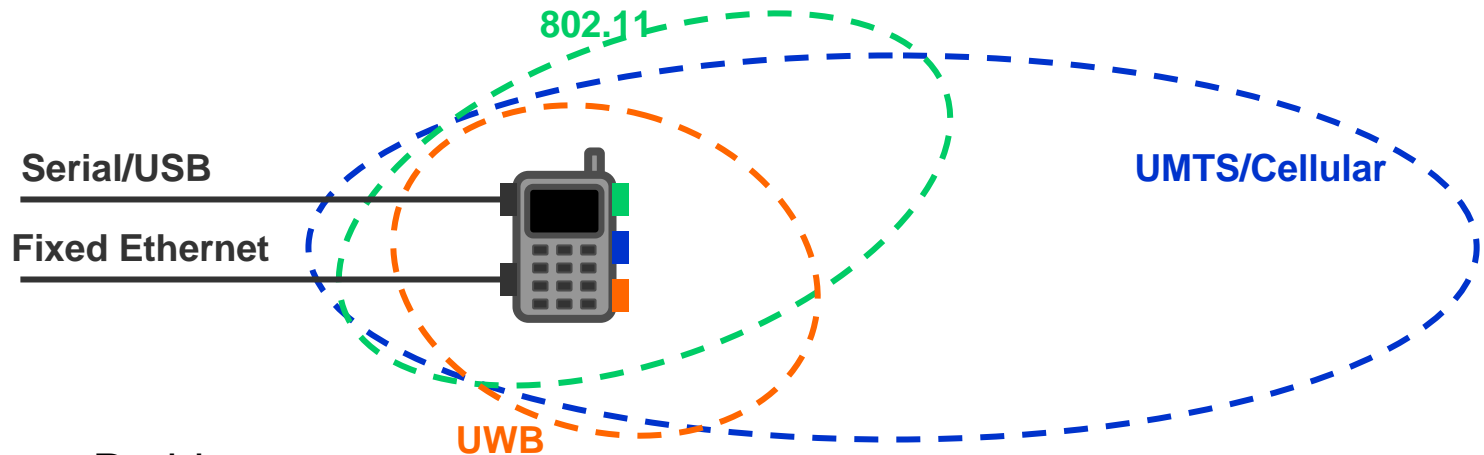
<sup>♦</sup>*INCA group / ‡AIMS group*  
*Telematica Instituut*  
*Enschede, The Netherlands*

<sup>\*</sup>*WMC group*  
*TU Delft*  
*Delft, The Netherlands*

SAACS'05, August 25, 2005

# The Initial (Concrete) Problem: *Adaptation to Network Resources*

- Reality today (and even more so in the future)



- Problem
  - What mechanisms to apply to make mobile host applications (and operating system) adapt to and exploit dynamic network resources?
  - Simple rule set won't do: high dynamics and heterogeneity in network technology results in a large (dynamic?) rule set for non-trivial adaptation

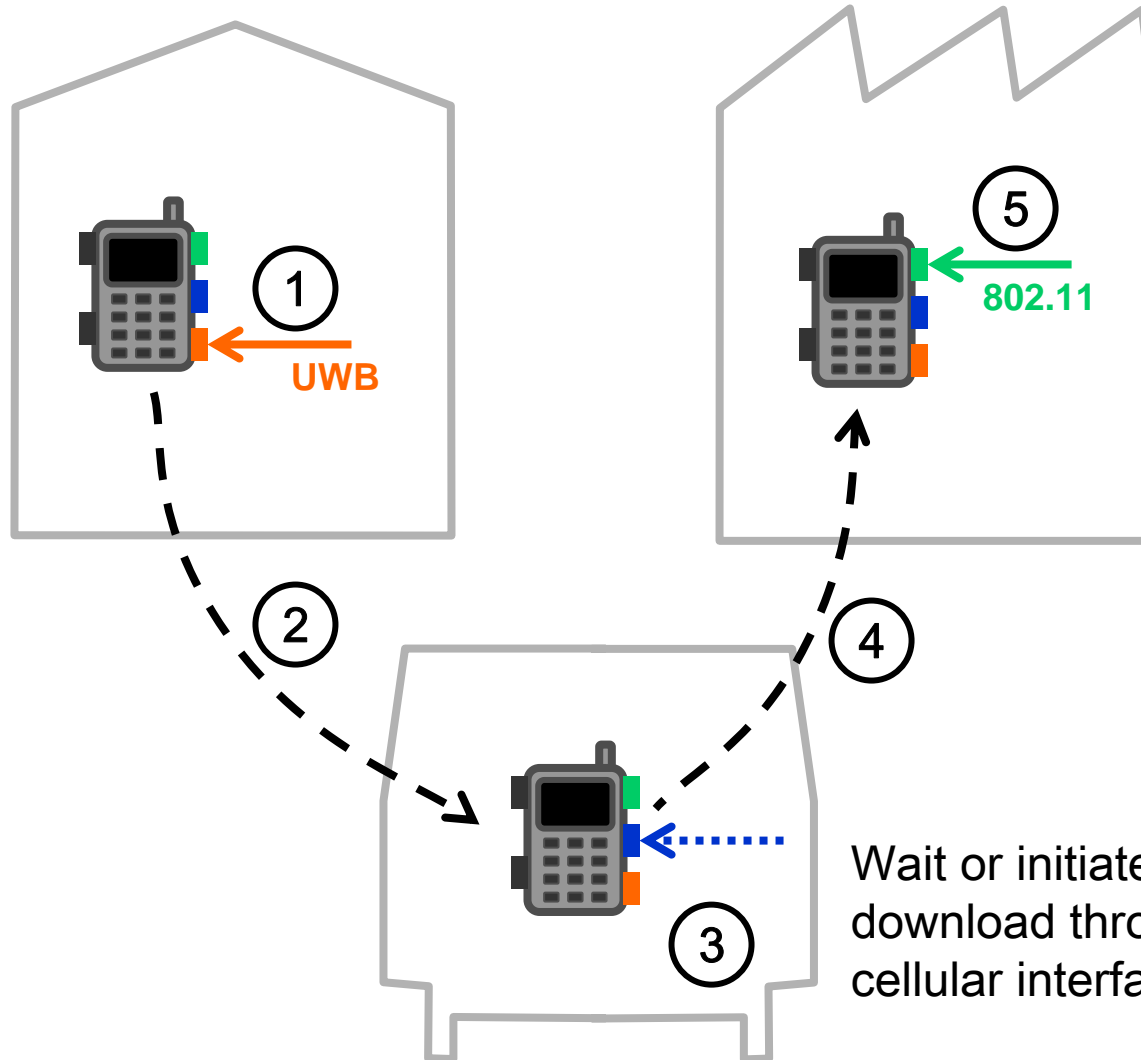


# Example of Network Resource Adaptation

- *Newsfeed application* that strives to have a local copy of the latest news multimedia content on the mobile host against minimal costs
  - User can watch latest news video
    - without waiting
    - when no cheap network available (e.g. in public transport)
  - Application needs to deal with various network characteristics (quality, capacity, cost of usage, ...) that is usually abstracted away by network and transport layer functionality
  - For better performance, the application may benefit from taking into account the user's previous behavior



# Example of Network Resource Adaptation

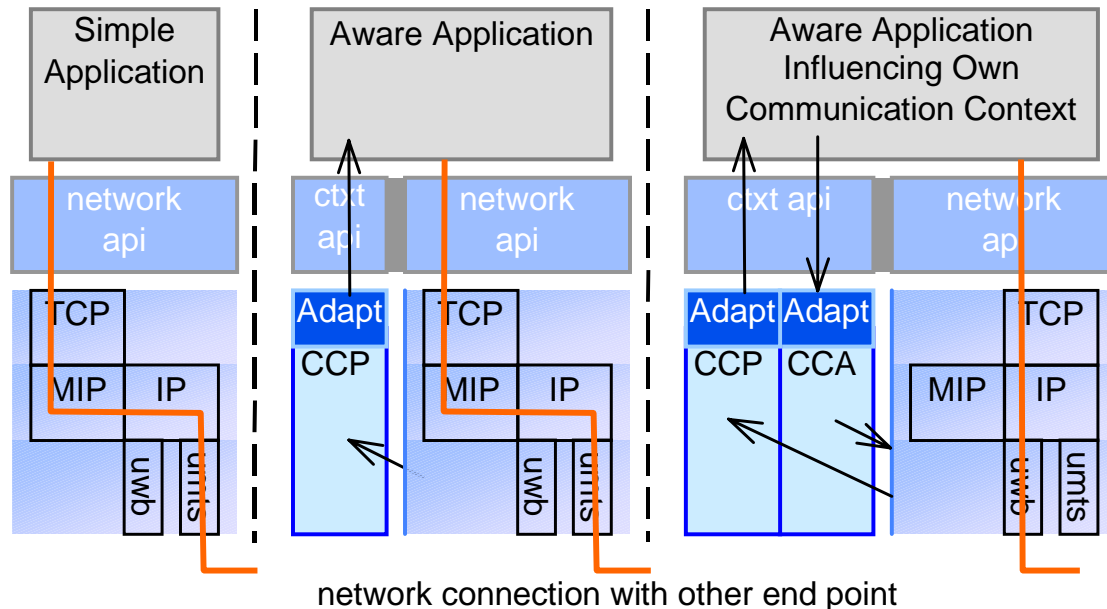


Wait or initiate  
download through  
cellular interface?



# Adaptation to Network Resources

- Part of the problem consists of supplying the application with the right and consistent (cross-layer) information about network and associated resources
  - We tackle this with the **CoSphere system facility**, capable of providing *Communication Context* to mobile applications (ongoing work\*)



\* See A. Peddemors, H. Eertink, and I. Niemegeers, "Communication Context for Adaptive Mobile Applications", In Proceedings of PERWARE'05, March 2005



## The Initial (Concrete) Problem: *Adaptation to Network Resources*

- However, taking a straightforward engineering approach, we still need to define the logic (program in code or whatever form) that determines the adaptation
  - E.g.: if mobile host in situation with wireless interface A in range of network K, and wireless interface B not in range of network L, then ... (etc)
  - Becoming highly complicated when taking a larger set of influencing parameters into account
    - User behavior history, time, location, ...
    - Especially when this set is dynamic!





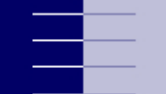
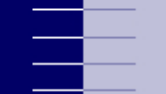
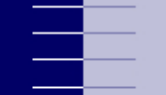
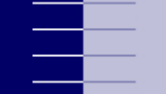
# The Initial (Concrete) Problem

## *... in the Light of Autonomic Computing*

- This is where Autonomic Computing and Autonomic Communication comes in...
- The adaptation to network resources strongly benefits from (taken from 'IBM's 8 elements')
  - Self-configuration and reconfiguration capabilities
  - Optimize own operation
  - Capability to know own environment and relevant context
  - Capability to operate in heterogeneous environment
  - Optimize resource usage and hide complexity
- Initial problem can be considered an autonomic computing problem
- So, let's look at the problem from a generic autonomic computing perspective
  - Topic of this presentation
  - **With a long term perspective in mind!**

# Outline

- Cognition and Autonomic Computing
- Towards System Support for Cognition
- Summary of Guiding Principles
- Contours of a Computer System with Cognitive Features
- Experiments and Applications
- Conclusions and Future Work



# Cognition and Autonomic Computing

- *Cognitive Capabilities* as essential ingredient for realizing 'IBMs 8 elements' for autonomic computing systems
  - Must "know itself", must be self-aware
  - Must self configure and reconfigure under varying and unpredictable circumstances
  - Continuously seeks ways to optimize own operation
  - Must be able to recover, to a certain extent, from malfunctioning ('healing')
  - Must be able to protect itself against various attacks
  - Must know its environment and relevant context for current activities and must be able to adapt to this
  - Must be able to operate in a heterogeneous environment
  - Must optimize resource usage and hide complexity



# Cognition and Autonomic Computing

## Definition of Cognition

The use and handling of *knowledge*; the mental process of knowing, including such aspects as sensation, perception, awareness, reasoning and judgement

- We know distinctive characteristics of Autonomic Computing from biological organisms only
- In nature, they are almost always related to *cognition*, except perhaps the self-healing ability
- In nature, cognitive capabilities are prerequisite for capabilities such as problem solving, self-awareness, attention
- Conclusion (taking the hint from nature): artificial cognitive processing functionality is a central element for autonomic computing



# Main Theme

Conceptual view on the incorporation of cognitive processing capabilities in future generation computer systems

- Based on two thoughts
  - Inspired by nature, we take the approach to mimic brain functionality with Artificial Neural Networks (ANNs)
  - We need to incorporate ANN functionality with existing system components
    - Hybrid approach mixing Von Neumann architecture with massively parallel ANN architecture
- Note: this is a conceptual work
  - Does not cover experiences with implementations or experimental results
  - Has elements that may not be realizable today



# Towards System Support for Cognition

- Implement basic principles and mechanisms that build cognition in nature
  - Not necessary to mimic the exact function of brain neuron: “only” the basic principles
- Biological cognition based on the *connectionist* principle
  - Neuron operation is simple
  - Network of neurons shows complex behaviour
- ANNs most likely need to be large for higher order cognitive capabilities (millions, maybe even billions)





# Towards System Support for Cognition

## *Complex Behaviour with Simple Rules*

- Current practice of building software applications\*
  - Implementation effort (programming) is intuitively proportional to the amount of functionality of the program
  - Highly complicated functionality (e.g. higher order cognition) is very complicated to build
- Neural networks
  - Simple rules, complex behaviour
  - Part of the effort shifting from *rule making* to *learning*
  - ANN only shows desired behaviour after sufficient training

\* As nicely illustrated by S. Wolfram, "A New Kind of Science", Wolfram Media, 2002

# Summary of Guiding Principles

- Bio-inspired approach when it comes to the implementation of artificial cognition
- Given large paradigm shift when going from a Von Neumann architecture to a connectionist architecture, we prefer a hybrid approach
  - Von Neumann model for traditional tasks
  - Parallel model for cognitive tasks
- Traditional parts setup with full control and monitoring by cognitive part
- Learning by observation



# Notes on the “Computational Metaphor”

- Computational Metaphor\*

Computation is a function from its inputs to its output. It is made up of a sequence of functional steps that produce – at its end – some result that is its goal.

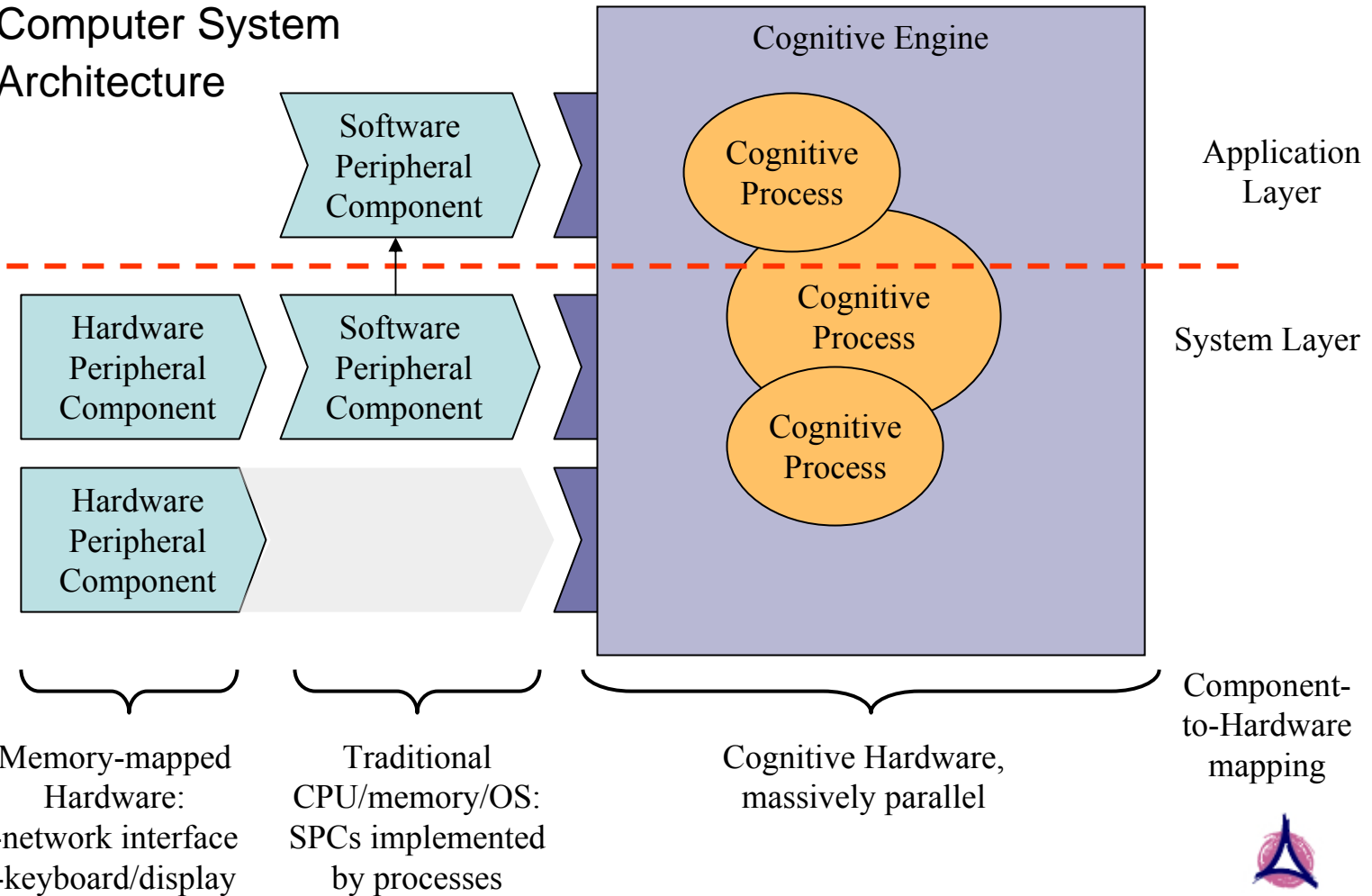
- Based on Turing’s or Von Neumann’s model
- Deviation from this model has far reaching consequences
  - We have all grown up with the Von Neumann machine and often use its principles in an implicit manner
  - Any structural change will be approached with scepticisms (and with good reason)
  - We tend to think of computation as a serial activity
  - Rich set of programming tools and many practical experiences are not usable anymore

\* As taken from L. Stein, “Challenging the Computational Metaphor: Implications for How We Think”, *Cybernetics and Systems*, Vol. 30, No. 6, pp. 473-507, 1999



# Contours of a Computer System with Cognitive Features

## Computer System Architecture



Memory-mapped Hardware:  
-network interface  
-keyboard/display

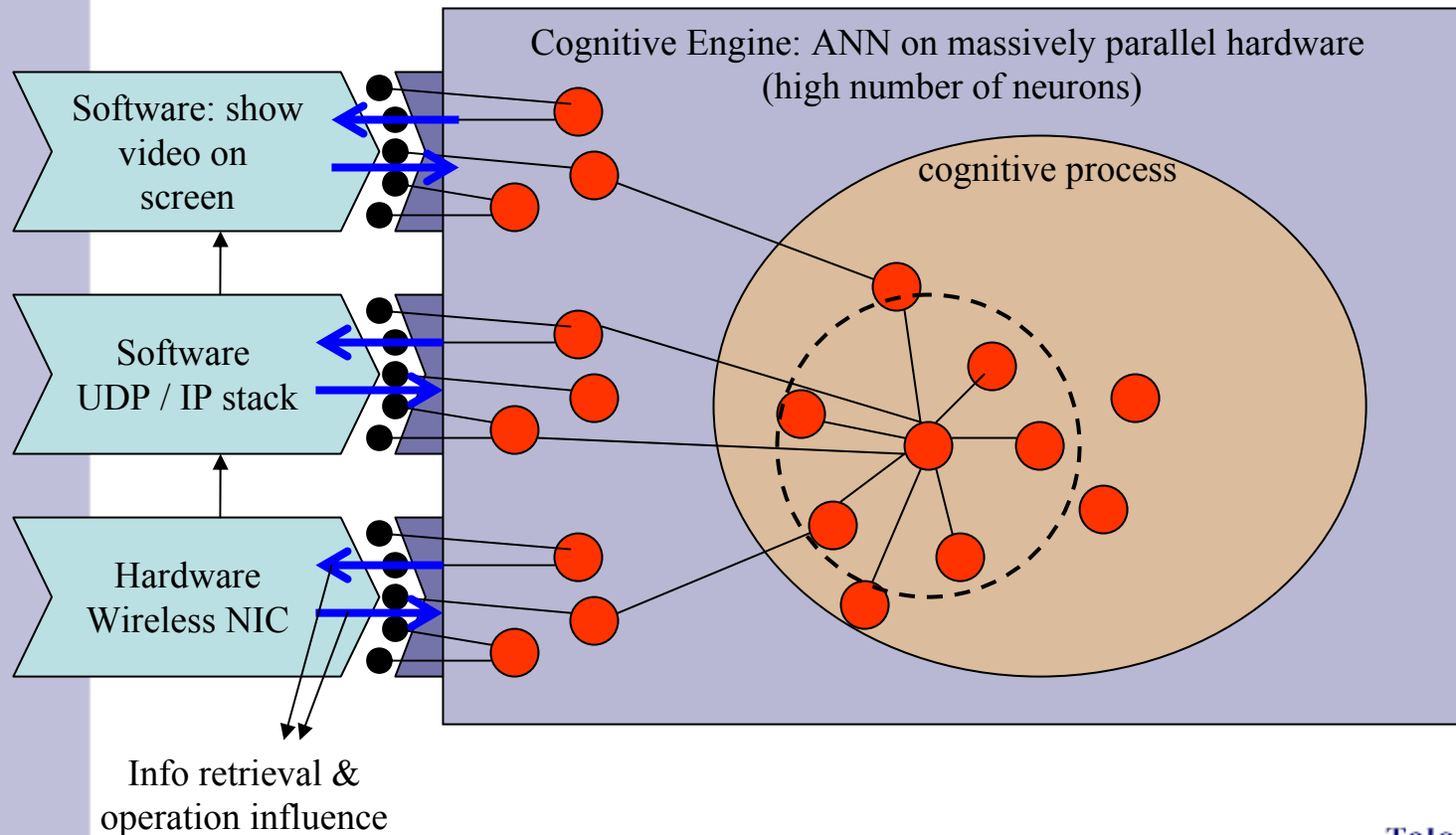
Traditional CPU/memory/OS:  
SPCs implemented by processes

Cognitive Hardware, massively parallel

Component-to-Hardware mapping

# Contours of a Computer System with Cognitive Features

- Networking example: all system elements involved in playing streaming video are controlled from the cognitive engine



# Some Notes on Mainstream CPUs

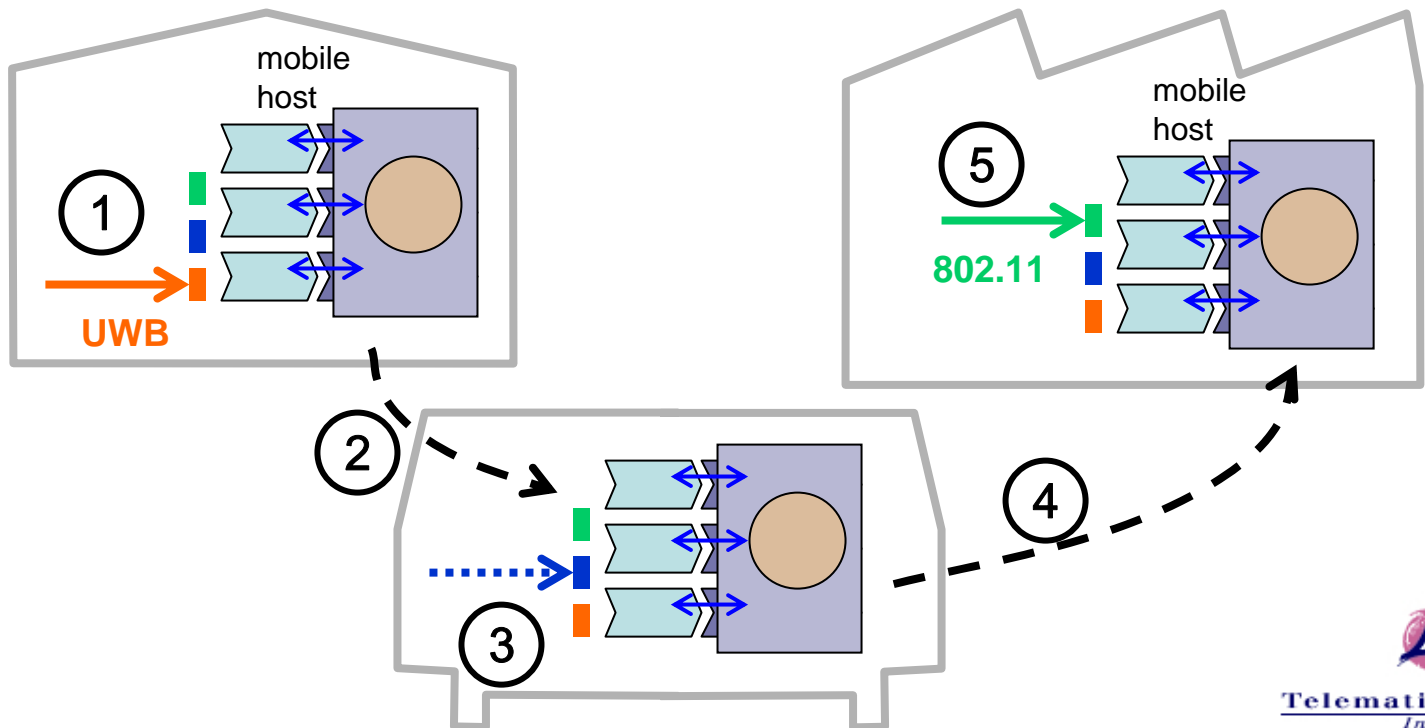
- Processing power gains
  - Less by increasing clock speeds
  - More by increasing parallelism
- X86 (Intel, AMD)
  - Dual-core processor on the market today
  - Announced plans to go to CPUs with four cores within a few years
- Cell (IBM)
  - Will soon ship to a mass market (Playstation 3)
  - Single Power core augmented with 8 co-processors
- More and more hardware parallelism
  - Easier to support real-time highly parallel ANN processing on standard computer equipment



# Experiments and Applications

## *Projection on the Initial Problem*

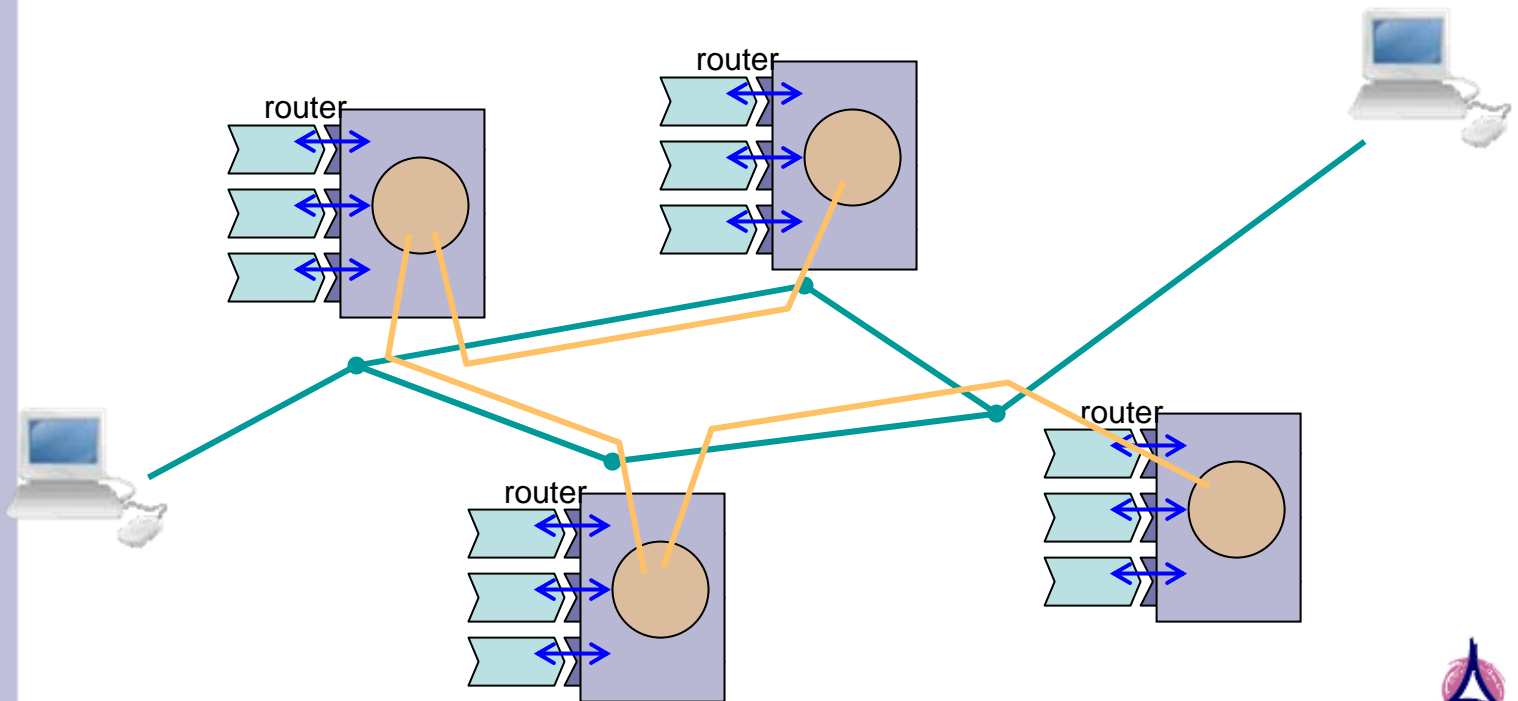
- Mobile host with hybrid system architecture
- Learning user behaviour
  - E.g. knowing that user, when leaving home in the morning, usually is quickly within reach of networks with plenty and cheap capacity (at the office)



# Experiments and Applications

## *Projection on Internet Knowledge Plane*

- The Knowledge Plane\* as self-regulating organism that controls the routing in the Internet
- Individual routers could be running on an ANN system architecture, where info is exchange between the cognitive cores of the individual routers

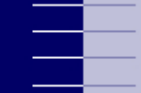
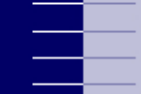
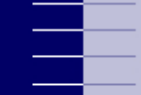
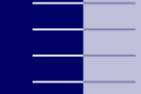


\* See D. Clark, C. Partridge, J. Ramming, and J. Wroclawski, "A Knowledge Plane for the Internet", In Proceedings of the ACM SIGCOMM, August 2003

# Conclusions and Future Work

- Provided a perspective on incorporation of cognitive processing features in future generation computers
  - Bio-inspired: ANN based cognition
  - Hybrid approach (mixed models)
- But, plenty of open issues
  - Still need progress on the ANN front
  - Hard to estimate the efforts necessary to learn the ANN
  - Mismatch between processing power and mobile devices
- Future work
  - Some initial experiments on feasibility of approach





Thanks!

Contact: [Arjan.Peddemors@telin.nl](mailto:Arjan.Peddemors@telin.nl)