



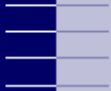
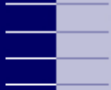
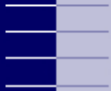
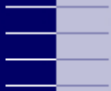
Communication Context for Adaptive Mobile Applications

Arjan Peddemors[†], Henk Eertink[†], and Ignas Niemegeers[‡]

[†] *INCA group, Telematica Instituut
Enschede, The Netherlands*

[‡] *WMC group, TUDelft
Delft, The Netherlands*

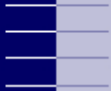
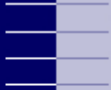
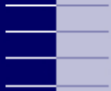
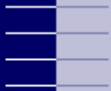
PerWare'05, March 8, 2005



Network Usage in Pervasive Computing *Mobile Devices*

- User surrounded by many artifacts and devices with computing and communication capabilities
- These artifacts and devices communicate over data networks
 - With each other or with other more remote services
 - In peer-to-peer, ad-hoc, or infrastructure fashion
- Typically, they are mobile from different perspectives
 - Relative to the user;
 - User leaving home or office
 - Vehicle (taxi) driving away from user
 - Relative to each other;
 - Music player moving around the house
 - Relative to data networks;
 - Personal wearable server in and out of reach of WLAN office network
 - HD Video Recorder in/out reach of user's PAN

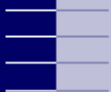
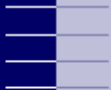
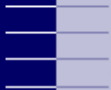
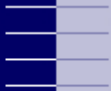




Network Usage in Pervasive Computing *Dynamic Network Environment*

- Given this mobility, many pervasive computing applications running on these mobile hosts have to deal with a **highly dynamic network environment**
- Furthermore, variations in network resources increased by
 - Usage of wireless network technologies: more dynamic characteristics than their fixed counterparts (interference, capacity variations,...)
 - Ability of many devices to simultaneously communicate over multiple network interfaces (often of different types)
 - Increasing heterogeneity in terms of wireless network technologies (continuous flow of new standards, ZigBee, UWB, WiMAX,...)



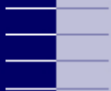
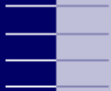
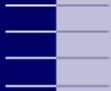
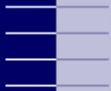


Network Usage in Pervasive Computing *Application Perspective*

How do applications on mobile devices perceive the available network and related communications resources?

- “Mode of usage” for peer-to-peer, ad-hoc, and infrastructure types of networks is often quite different
- Mobile applications take into account, implicitly or explicitly, the specifics of these modes
 - For example, actions concerning network activation:
 - P2P: application activates link when necessary
 - Infra: often regarded by applications to be “simply there”
- Unfortunately, a unifying mechanism for the communication of applications with other nodes, close or remote, in a heterogeneous network environment (p2p, ad-hoc, infra), does not exist

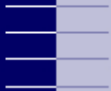
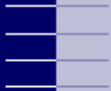
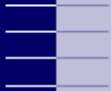
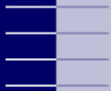




Network Usage in Pervasive Computing *Application Perspective*

- OK, some believe the Internet Protocol may fit this role (“every conceivable entity its own globally unique IPv6 address”), but
 - Requires, obviously, usage of IP for all communication
 - Too heavy or inappropriate for certain types of local communication
 - Think: low power sensor networks
 - Think: dedicated peer-to-peer mechanisms such as found (*and widely used*) in Bluetooth; Object Exchange (OBEX) protocol, etc.
- As a consequence, there is no single mechanism and abstraction that can be used for **all** network communications of pervasive computing (mobile) applications

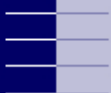
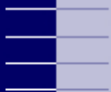
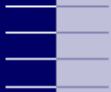
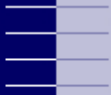




Network Usage in Pervasive Computing *Application Perspective*

- By itself, this is not a problem; it is a reflection of the fact that different types of networks have different modes of usage
- Having said this, we feel that for many mobile applications, IP-based communication is (and will be) most widely used
 - Certainly for infrastructure networks
 - Possibly also for peer-to-peer and ad-hoc networks
- Taking IP as a starting point, we have a layer that is abstracting away all technology specific information of link an physical layer
- Network resources are, however, very dynamic and applications need to adapt to variations in network resources but do not see changes due to the IP abstraction

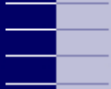
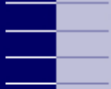
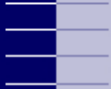
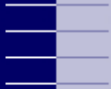




Objectives

- Provide information about network resources and related communication facilities to applications running on a mobile host
 - Assuming IP-based communication
 - In a heterogeneous network environment
- Provide this information in such a way that it is easy for mobile application to adapt to network resource changes
- Allow mobile applications to influence the available network resources (e.g. influence 802.11 activation, etc.)

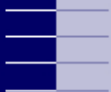
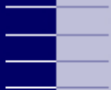
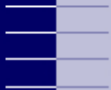
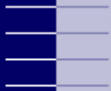




Outline

- Communication Context
- CoSphere Architecture
- Application Interaction
- Summary and Future Work





Communication Context

Introduction

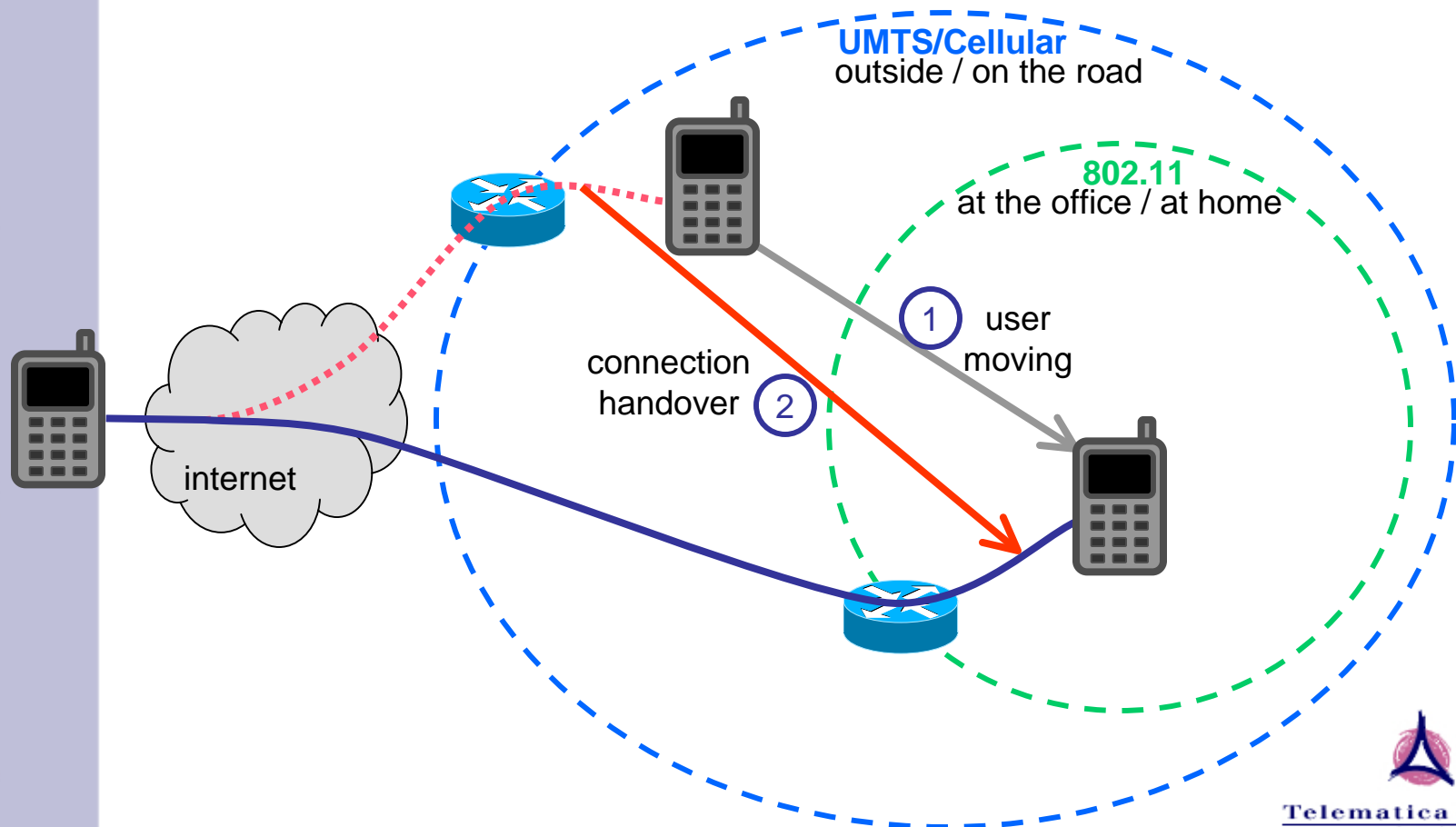
- Relevant aspects for communication of mobile applications
 - *Network Context*: current state and characteristics of available networks on mobile host (cross-layer information)
 - *State of (operating) system facilities* relevant for communication (e.g. network selector functionality, mobility management protocols)
 - *State of network infrastructure facilities* for communication (e.g. DNS and proxy facilities)
- Information on these combined aspects defined as **“Communication Context”**
- Note: communication context expresses more than just available network resources

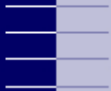
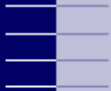
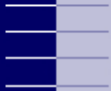
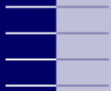


Communication Context

Example: Voice over IP

- Mobile device has two network interfaces
- VoIP application running on mobile device





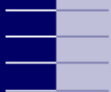
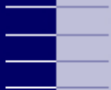
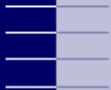
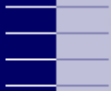
Communication Context

Example: Voice over IP

Communication Context aspects relevant for VoIP application:

- *Network Context (cross-layer information)*
 - Which networks are currently actively accessible through the mobile device?
 - What are the characteristics of the available networks?
 - Throughput capacity, delay parameters, quality indicators
 - If currently UMTS only, indicate interest in discovering 802.11 network
 - For power consumption reasons not all network interfaces are in an active state at all times
 - Note: application takes into account characteristics such as range. Effectively falling back on cellular network when no wlan; implicitly assuming high UMTS availability
 - If “office” or “home” 802.11 network in range, indicate interest to actively connect to one of them (not just any available 802.11 network; need to know 802.11 network identity which is link layer info)





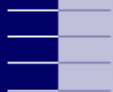
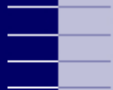
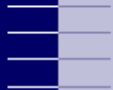
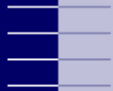
Communication Context

Example: Voice over IP

Communication Context aspects relevant for VoIP application (continued):

- *State of relevant system facilities*
 - Path selection: over which path is my connection traffic directed by the OS?
 - Change in path => change in used network interface => change in network technology => change in network characteristics (throughput, delay, packet loss,...)
 - Are there any upcoming changes in the availability of current networks?
 - What facilities or protocols are available to manage the connections in case of a network handover (from UMTS to 802.11 and vice versa)
 - Mobility management protocols such as Mobile IP, but also Mobile SCTP, HIP
- *State of relevant infrastructure facilities*
 - Where can I find out where my local VoIP call session setup facilities (e.g SIP) are (local to currently activated networks)?



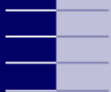
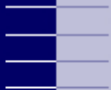
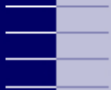
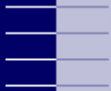


Communication Context

More examples...

- Adaptive Streaming Media Application
 - Adapting transfer rate according to available network resources (which are dynamic in mobile / roaming environment)
 - Voice over IP
- News Feed Application: bursty network traffic
 - Occasionally shows high-rate downloads when plenty of free network capacity is available
- Healthcare Sensory Application
 - Typically organized as cooperating entities in Body or Personal Area Network (BAN/PAN): distribution aspects of communication context
 - Adapt behaviour of uploading of sensory data to medical center depending on communication context



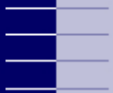
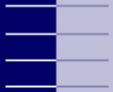
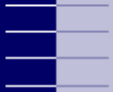
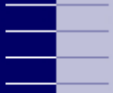


Communication Context

Levels of Abstraction

- The previous examples show that the communication context is useful to mobile applications that need to adapt to dynamic network resources
- However, it is also clear that some applications need more detailed information than others
 - Ranging from no more than the abstraction offered by plain IP (or better, transport layer protocols such as TCP and UDP on top of IP)
 - ...to highly detailed technology specific information about physical and link layer characteristics
- Therefore, we propose to provide communication context to applications with different levels of abstraction
 - Giving them just the right amount of information to be adaptive (no more, no less)
- At each abstraction level, communication context is linking information that previously was separated (or non-existent)
 - Network characteristics to network interface identifier to IP configuration to mobility management protocol state to path selection (routing) to transport layer connections

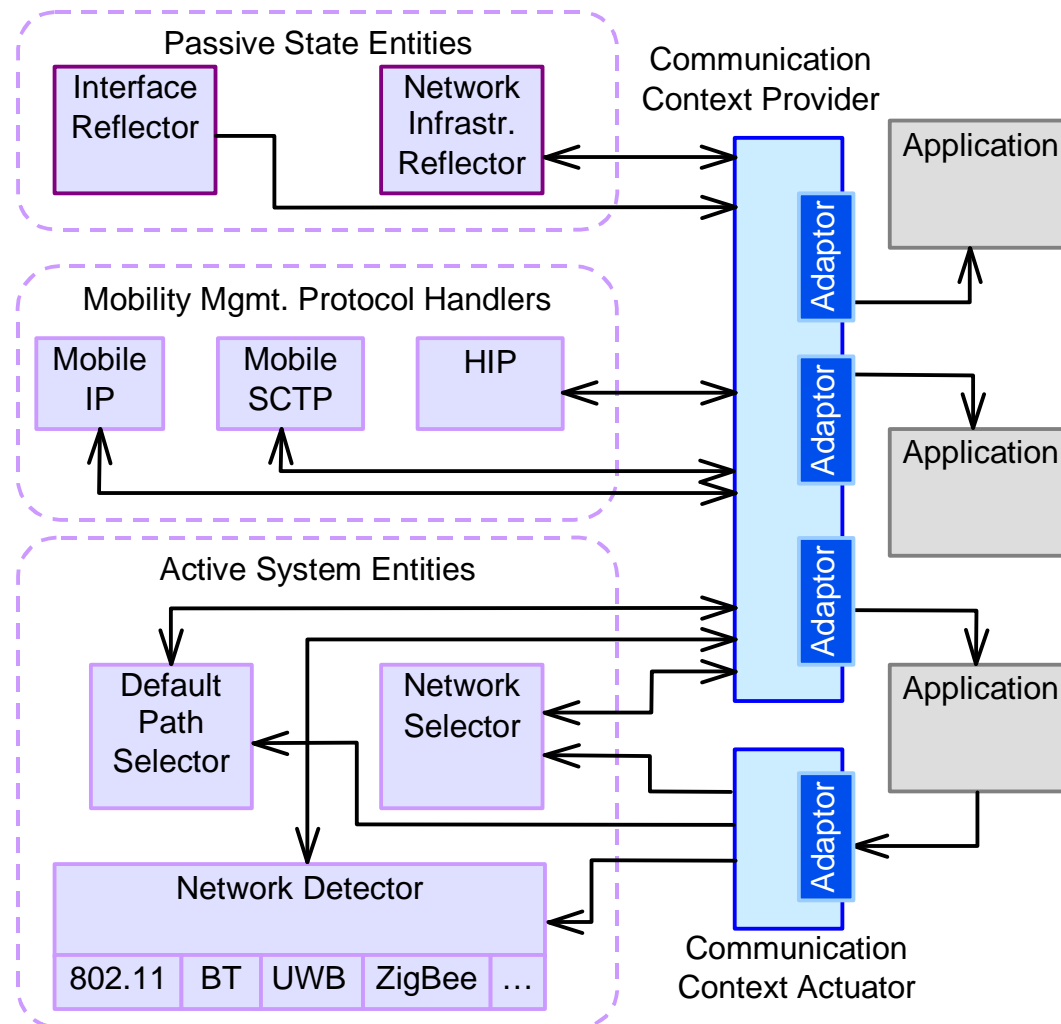


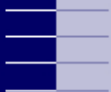
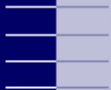
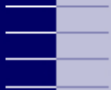
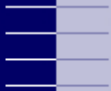


CoSphere Architecture

- We are currently working on a system level facility, called *CoSphere*, available on mobile hosts
 - Responsible for providing communication context to application running on the mobile host
 - Integrating existing and new system components to be able to generate a coherent communication context
 - Controlling and integrating with Operating System network functionality
 - Allowing application to influence their own communication context (e.g. indicate interest in the activation of certain networks)
- As IP has a central role in our concepts and architecture, the system service has a bias toward infrastructure networks, but could also be applied to peer-to-peer and ad-hoc networks
- Partly based on previous work dealing with application aware mobility management (current work has a broader scope)

CoSphere Architecture



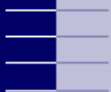
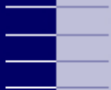
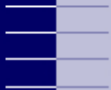
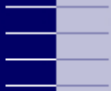


CoSphere Architecture

Components Interacting with Applications

- Communication Context Provider (CCP)
 - Defines a number of different levels of abstraction and detail, targeted at a specific type of application (implemented by *adaptors*)
 - Application chooses adaptor when connecting
 - Identification and definition of the different abstractions is a work in progress
- Communication Context Actuator (CCA)
 - Allows application to influence their own communication context
 - Adaptors allow manipulation at different levels of abstraction
- CCP + CCA = *Communication Context Service (CCS)*, running as a background process (“daemon”) on the mobile host





CoSphere Architecture Components “under the Hood”

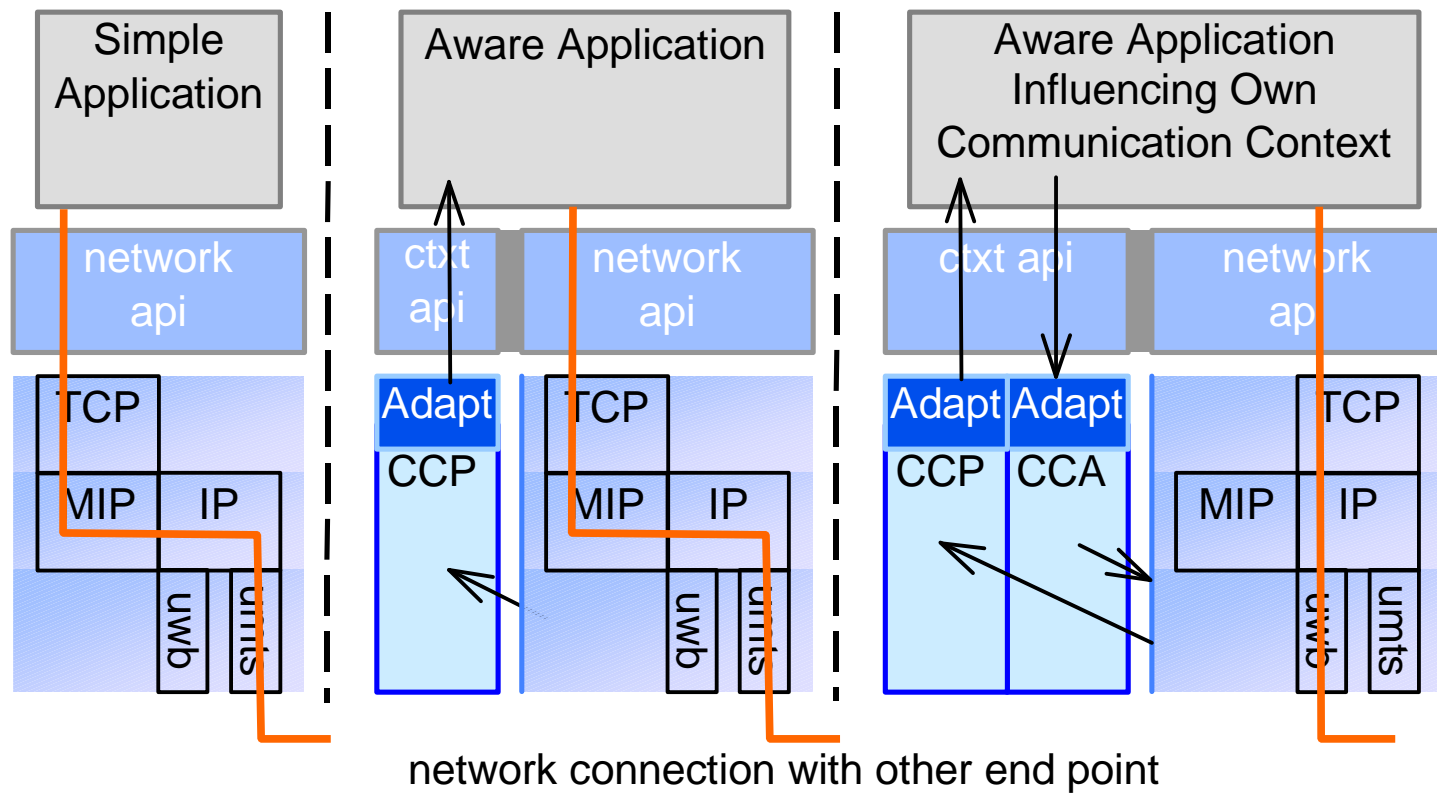
All of them are ‘raw’ context sources; the CCS collects, aggregates and infers the data for the applications

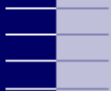
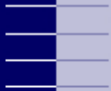
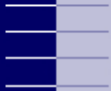
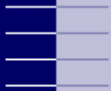
- Active System Entities
 - Influence communication context with their actions
 - Detection of networks (802.11 scanning, BT discovery, etc.)
 - Network selection; multiple networks may be available through an interface but often only one can be made active (cellular, 802.11)
- Mobility Management Protocol Handlers
 - Special form of active system entities, with focus on providing mobility support (in different forms)
 - Application may choose mobility management protocol depending on required characteristics
- Passive State Entities
 - Do not influence the context; just provide information



Application Interaction

Different types; different interactions



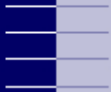
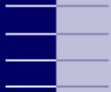
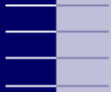
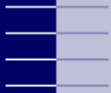


Application Interaction

Different types; different interactions

- Different types of applications interact with the Communication Context Service in different manners
 - Simple application
 - “Traditional” application; using plain IP (+ standard transport layer protocols)
 - Aware application
 - Obtaining communication context through context API
 - Aware application influencing own communication context
 - Obtaining and influencing communication context through context API
- Applications using different levels of abstraction of the communication context provided to them, depending on level of detail needed
- Context API co-exists with standard network API (Sockets API)

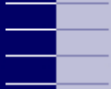
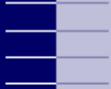
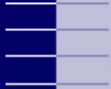
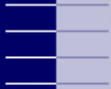




Summary and Future Work

- Summary
 - Presented communication context and motivated its usage by applications running in highly dynamic network environment
 - Introduced the CoSphere architecture, capable of generating communication context at different abstractions levels, allowing application influence
- Future work / in progress
 - Detailed analysis of application information needs so that they may be adaptive towards available network resources
 - Provide a reference implementation of the CoSphere architecture
 - Validate the architecture using a number of different types of applications in a non-trivial heterogeneous network environment
 - Explore possibilities to distributed communication context in short-range networks (PAN)





Thanks!

Contact: Arjan.Peddemors@telin.nl