

Towards a service definition for (Wireless) Sensor Networks

Prof. Dr.-Ing. Adam Wolisz

www.tkn.tu-berlin.de

Zürich, April 1st, 2004



Telecommunication Networks Group
Technische Universität Berlin

Explicit Acknowledgements are due to

- Co-authors of the white paper

“A Service-Based Universal Application Interface for Ad-hoc Wireless Sensor Networks”

By: Marco Sgroi, Adam Wolisz, Alberto Sangiovanni-Vincentelli, Jan M. Rabaey

- The TU Berlin/TKN Staff!
- PicoRadio Staff at UC Berkeley/BWRC

Sensor networks for content capturing and more

- Today's information systems care mostly about distributing information (WWW) with lousy support for gathering information (crawlers), slightly better for selection (Google) and no support for actuation.
- Physical world has to be sensed and **controlled!**
- We **DO** have nodes - and will have better and better each year (month?).

Crossbow, Intel, Infineon (CSEM; IMEC; ETHZ) etc. will start selling them for few Euros/dollars/rupia...

So what??

Typical scenarios for the sensing side...

- **Data Collection:** with a frequency specific for the physics of the plant. Some aggregates: Like averages over short time and pre-defined space are expected.

Long term operation (years!) is crucial!

- **Alarm monitoring:** Pre-defined events have to be identified and signaled ASAP. Permanent self-testing of the sensors required.

Reliability of the operation and short delay are crucial!

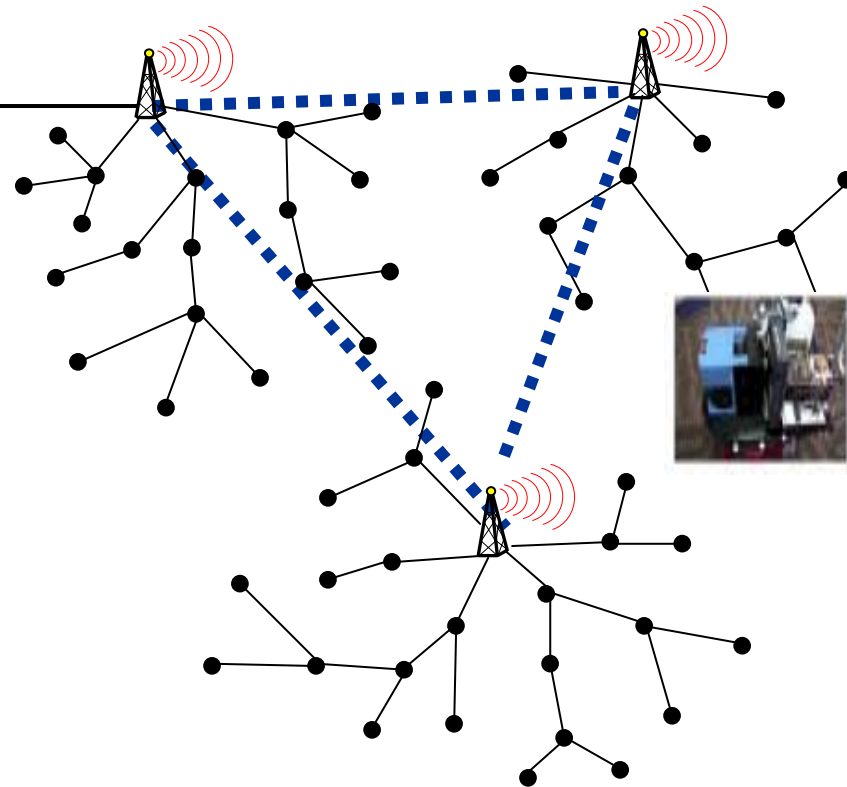
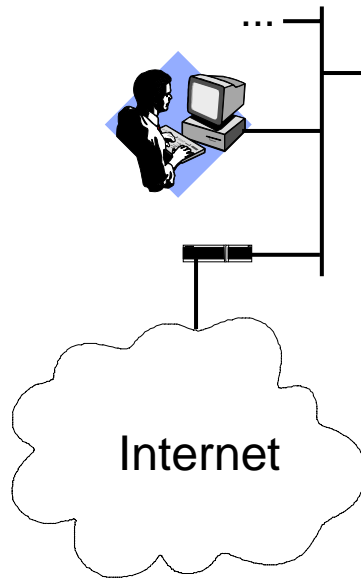
- **Object tracking:** Objects (people, cars) emitting some specific values have to be tracked.

Position estimation in spite of changing dynamics...

Sensor Network...

Classical Infrastructure

Sensor Network



Sensor Node



Low Power Radio Link



Base Station



High Bandwidth Radio Link/fixed Link?

It is all about service **interface!**

- Focus is on the information (or the service), not on the device or their interoperation.
- **Nodes** might be different, **networks** differently organized.
- Nodes might be on/off...
- A BIG gap between the applications requesting some information and some node (potentially!) contributing to its computation.

Standardization of services and APIs
NOT unification of solutions.

Significance similar to the significance of the socket interface in the Internet development.

Sensors – Actuators - Controllers

- *Controllers: Data processing (active component)*

- *Sensors: Data Sources (reactive)*

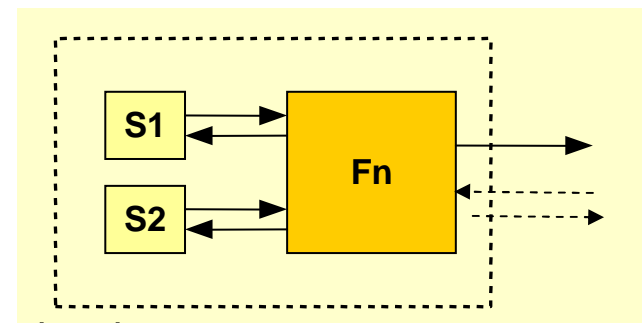
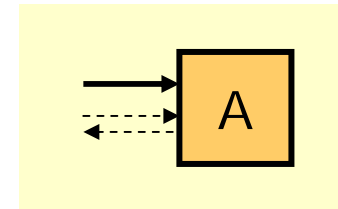
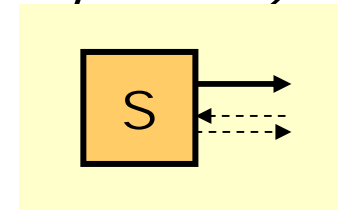
- Report the state of the Environment

- *Actuators: Data Sinks (reactive)*

- Set the state of the Environment

- *Virtual Sensors/Actuators*

- a group of (sensors + controllers)



Parameters of sensors/actuators:

- Examples: range, accuracy, sampling rate, date of calibration, past measures available
- *In the style of the IEEE 1451.2 standard*

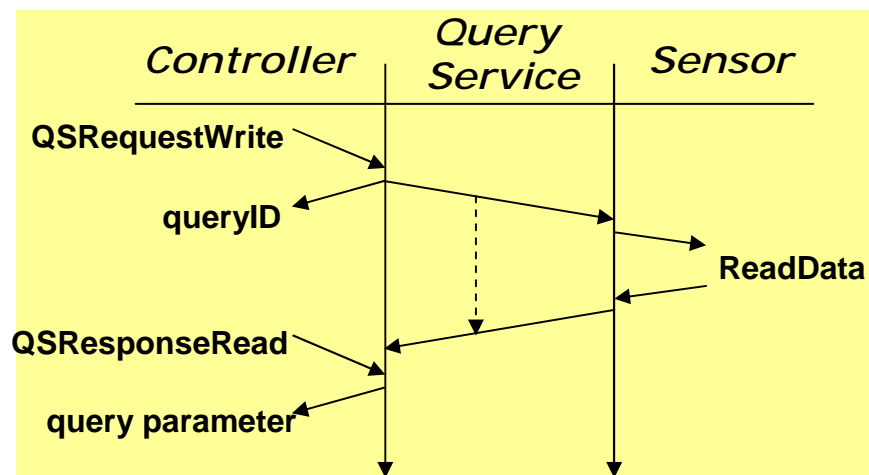
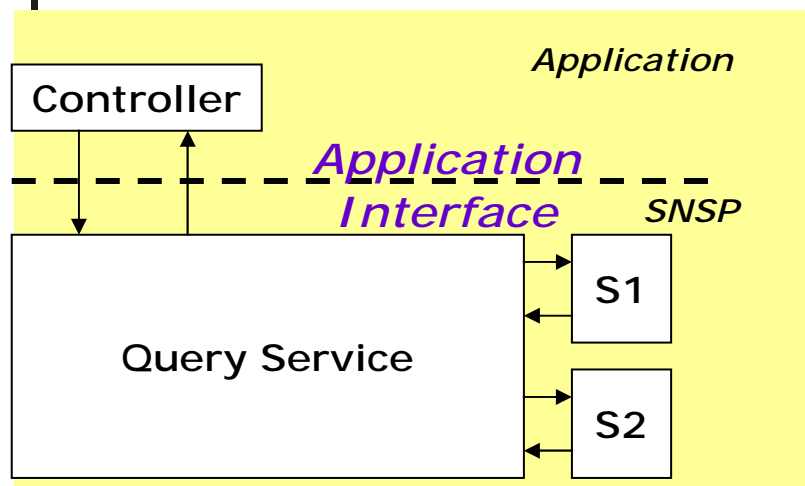
Semantic addressing

- **Interest in** : temperature in the Kitchen
NOT referring to node A20 (or 10.32.117.241)
- Address (not unique!!) consists of a pair:
 - **Attribute specification** : Attribute and value selector
((temperature, > 25 degrees) OR (humidity, = 70 %))
 - **Scope**: description of an *organization* or *region*
Org. : Utility company (electricity provider), Fire Dept..
Region: Kitchen, courtyard; **or**: 30 meters around XXX
- **How to? Who cares!**

*Possibly by flooding the request and collecting the answers
(see for example Directed Diffusion). Possibly not....*

- **Quantifiers** (Semantic Addresses are NOT unique!)
 - Examples: *all*, *at least k*, *exactly k*, *any*.
- **Response Types:**
 - *one-time* response (in answer to the request)
 - *periodic* responses with interval period p
 - *notification* of events whenever an event occurs
- **Other query parameters:**
 - *Operation* (like average value, max value)
 - *Accuracy* (required from the answer)
 - *Timeliness* (requested)
 - *Tagging* (should the answer contain time and location)
 - *Security* (which level is requested)

Query Service - explanation



QS allows a controller to obtain values computed on the basis of the state of a group of components

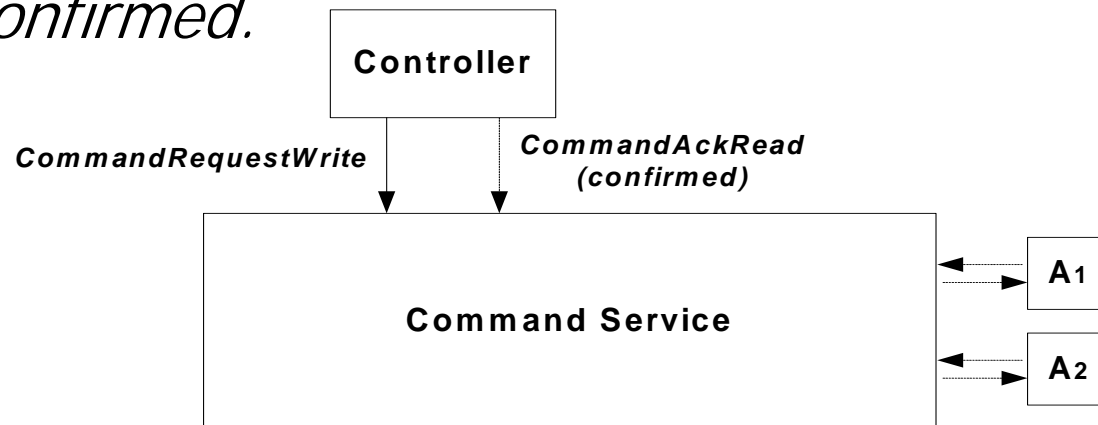
Commands for actuators work in a mirrored way

In Network Processing – in the background

- Addressing supported by in-network processing of the semantic address description
- Sensing the physical reality implies correlation of data in space and in time, also limited gradient of changes...potential for data reduction!
- Query processing NOT equal to sensor reading!
 - Provide possibly already stored data (if multiple requests)
 - Aggregate data (forward only ONE out of several readings)
 - Compute functions (with given accuracy) in the network!
- **But, again: this is a competitive area of HOW... Applications have to be portable across the solutions!**

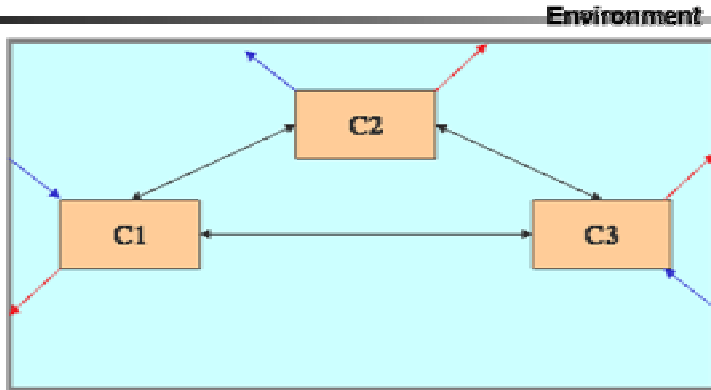
Commands...

- Initiated by a controller (*command initiator*), which demands a group of components (*command targets*) to take an action. The command is of type *confirmed* or *unconfirmed*.

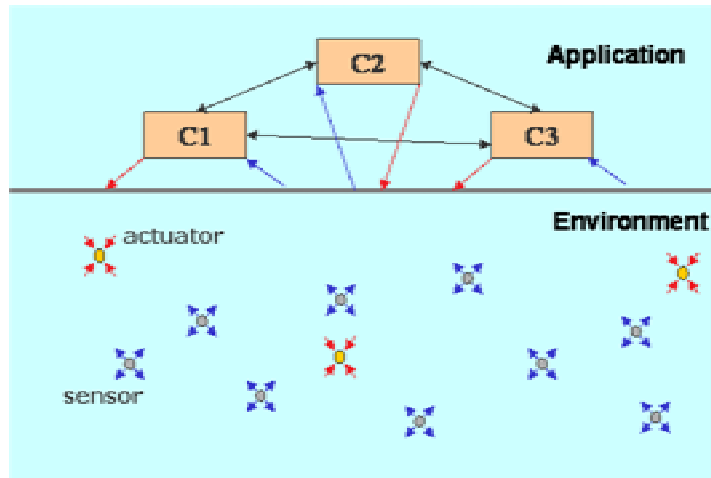
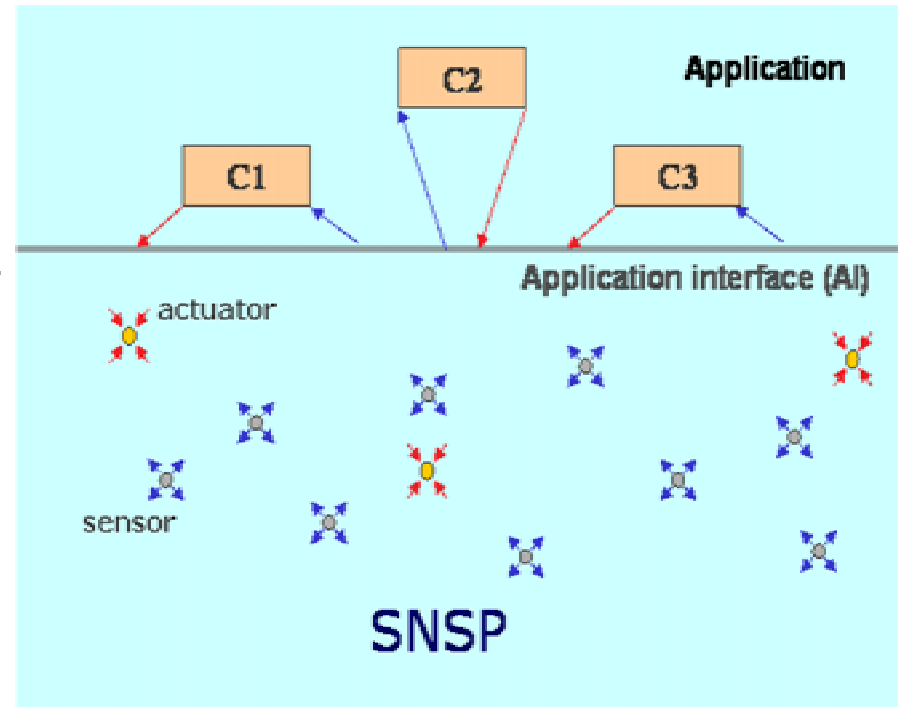


Parameters: Accuracy, resolution, security, priority, quantifier, operation)

Communication among controllers...



(a)



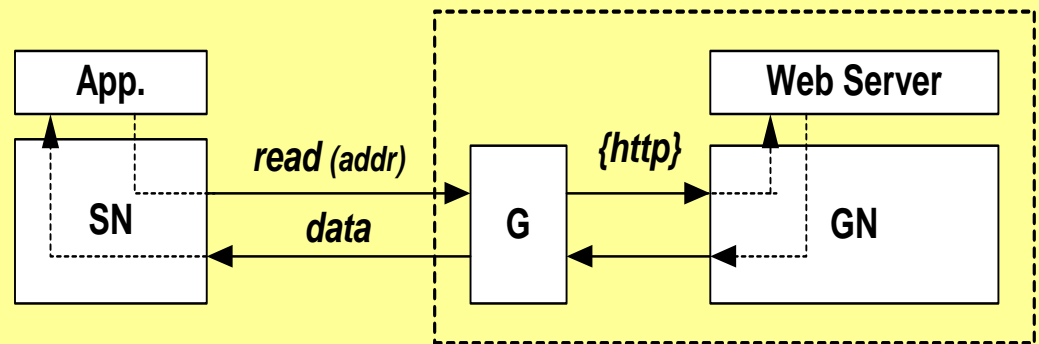
(b)

Controllers **expose** some information (to other controllers) like sensors! Controllers set (**exposed**) values of other controllers like they set actuators! CLEAR INTERFACES!!

Networks Interoperation – Virtual Sensors

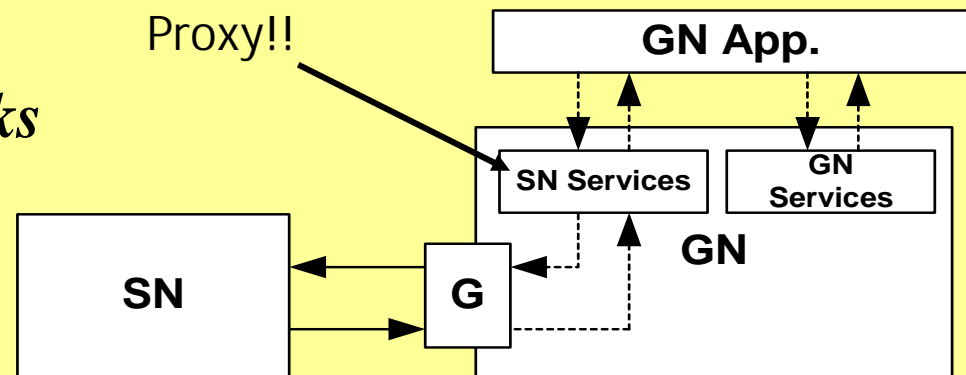
Read the forecasted temperature for Zürich, 3.04.04

Access to Global Services



Get the temperature in the kitchen to my PDA

Accessing SNs from Global Networks



Additional services (**interfaces** NOT solutions!)

- Location service:

What is my location (in agreed coordinates, with given accuracy)?

Define borders of a region (kitchen, city district) and neighborhood (distance from) In which region am I?

- Time synchronization service

For a given neighborhood, with a given accuracy...

Other might be included

- Consensus Service: how to make the controllers agree on some values?

Time Synchronization Service

TSSActivateSynchronization

TSSGetTime

TSSSetTimer

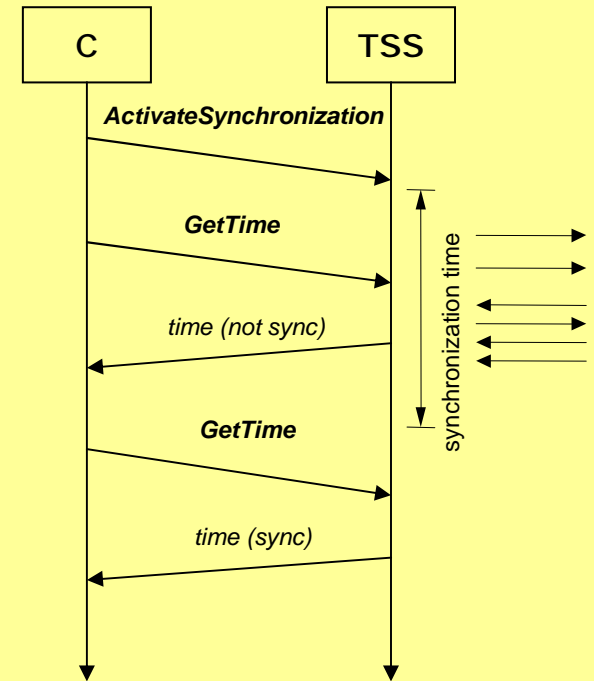
Parameters: Synchronization Scope/reference

Resolution

Accuracy

(Security)

(Availability)



TSS allows two or more system components to share a common notion of time and agree on the ordering of the events that occur during the operation of the system

- Within an instantiated node components share time (same clock)
- Node can be synchronized with other nodes or use local time

Semantics: The Concept Repository Service (CRS)

- CRS maintains a repository containing the lists of capabilities of the network and the concepts that are supported. *?? HOW TO DO IT ???*
- Concepts
 - Attributes (used for naming)
 - Regions (zone, neighborhood)
 - Organizations
 - Selectors, Logic operators, Quantifiers
- Repository updated by the application or automatically (plug-and-play)
- Allows to maintain agreement on concepts also in dynamic network operation

Hardware abstraction in WSN operating systems

- What is the most appropriate level of *hardware abstraction* ?
- Two seemingly conflicting requirements
 - Rapid application development → high level of abstraction
 - Energy-efficient implementation → low level of abstraction
- How to effectively reconcile this gap ?
 - Flexible, reconfigurable operating system architectures
 - More like a component library than a monolithic structure
 - Let the application choose the appropriate level of abstraction
- But: **let us express this in a unified way...**

Management Service

- Uses the Query/Command mechanism for getting/setting values pertaining to the hardware, e.g. the level of the battery charge.
- No new mechanisms but new way of making the proper values available.
- Management Applications (controllers) set the modes of operation according to specific operation policies...

Thank You !



Questions?
Ask now!

Alternatively :
Talk to me later
or drop me an
e-mail: awo@ieee.org