



# Open challenges in ubiquitous and net-centric computing middleware

[bodhuin/canfora/preziosi/tortorella@unisannio.it](mailto:bodhuin/canfora/preziosi/tortorella@unisannio.it)

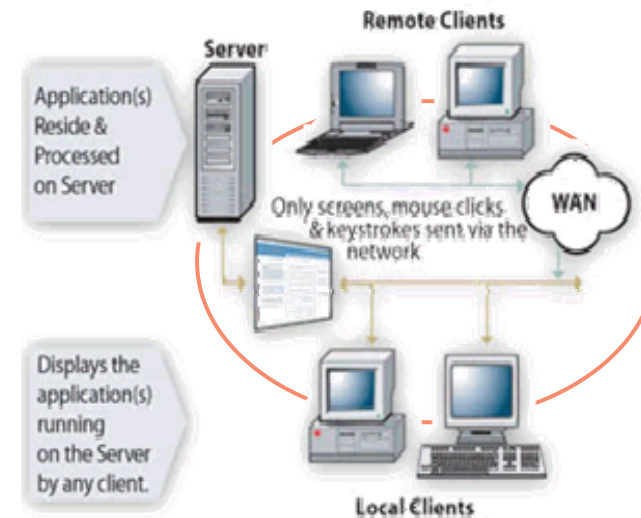
RCOST – Research Centre On Software Technology  
University of Sannio – Benevento, Italy

# Net-Centric Computing (NCC)

The underlying principle behind NCC is a distributed environment where applications and data are downloaded from network servers on an as-needed basis.

This is similar to the mainframe era, except that applications are downloaded and then run locally.

This style of computing relies on portable applications, high bandwidth (at least 10 Mbps, preferably 100 Mbps in a local set-up), and low-cost thin clients.




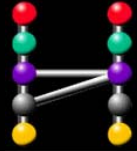
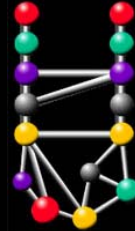
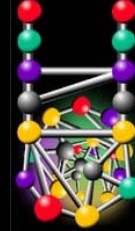


# Net-Centric Computing (NCC)

Web, Java, RMI and CORBA are traditional solutions for NCC. Together they offer a complete distributed framework in support of net-centric applications. Unfortunately these solutions alone do not permit the life cycle management of services, are protocol dependent, not enough dynamic, ....



# NCC Platform Evolution

Catch Phrase	The Network Is the Computer	Objects	Legacy to the Web	The Computer Is the Network	Network of Embedded Things	Network of Things
Scale	100s	1,000s	1,000,000s	10,000,000s	100,000,000s	100,000,000 s
When/Peak	1984/1987	1990/1993	1996/1999	From 2001	Since 1998	From 2006 ?
Leaf Protocol(s)	X	X	+HTTP (+JVM)	+XML Portal	+RMI/JRMP =>+RMI/JERI	+RMI/JERI ?
Directory(s)	NS, NS+	+CDS	+LDAP	+UDDI +ebXML	+Jini Lookup service	+JXTA + Jini?
Session	RPC, XDR	+CORBA	+CORBA, RMI	+SOAP, XML	+RMI/Jini	+SOAP, XML ? +Jini ?
Schematic						



# Ubiquitous Computing (UC)

## **Ubiquitous computing (or pervasive computing)**

integrates computation into the environment, rather than having computers which are distinct objects.

One of the goals of ubiquitous computing is

- to enable devices to sense changes in their environment and
- to automatically adapt and act based on these changes based on user needs and preferences.

Some simple examples of this type of behavior include [GPS](#)-equipped automobiles that give interactive driving directions and [RFID](#) store checkout systems.

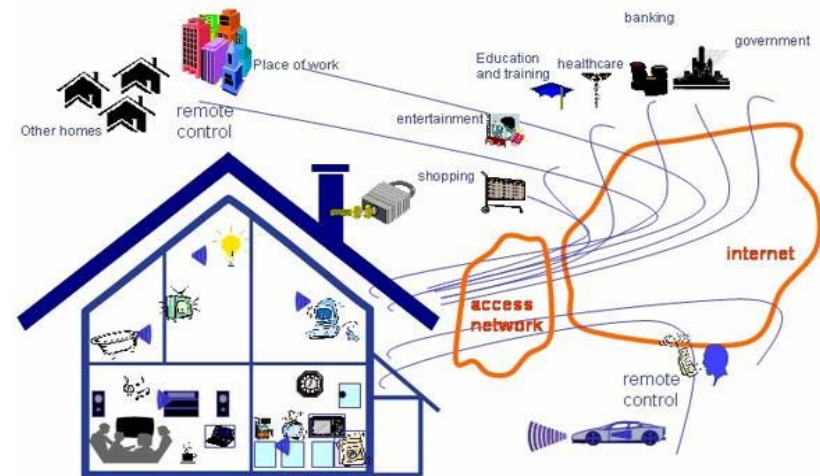


# Ubiquitous Computing (UC)

A revolution in the field of computation, communication and interaction is taking place, and Weiser's vision described in 1991 is becoming a reality.

Protocol	Note
Batibus	Widely used in France. <a href="http://www.batibus.com/">http://www.batibus.com/</a>
Bluetooth	In an expansion stage, but broadly supporting short-range wireless technology. <a href="https://www.bluetooth.org/">https://www.bluetooth.org/</a>
EIB	(European Installation Bus) with home electronic system (HES) user interface: number 1 in Europe. <a href="http://www.eiba.com/">http://www.eiba.com/</a>
Ethernet, TCP/IP	Widely used in the field of IT.
i-Link	(fire wire, IEEE 1394). Multimedia bus systems. <a href="http://www.ieee.org/">http://www.ieee.org/</a>
LON	(Local Operating Network) from Echelon: widely extended in America. During the last years it is more and more accepted in Europe. <a href="http://www.echelon.com/">http://www.echelon.com/</a>
X-10	Primarily used in the USA <a href="http://www.x10.org/aboutx10.html">http://www.x10.org/aboutx10.html</a>

- the world will consist of ubiquitous systems rather than “*the ubiquitous system*”;
- large, highly dynamic distributed systems, always working and available, will be conceived;
- “*browsing reality*” will be similar to browsing the web.

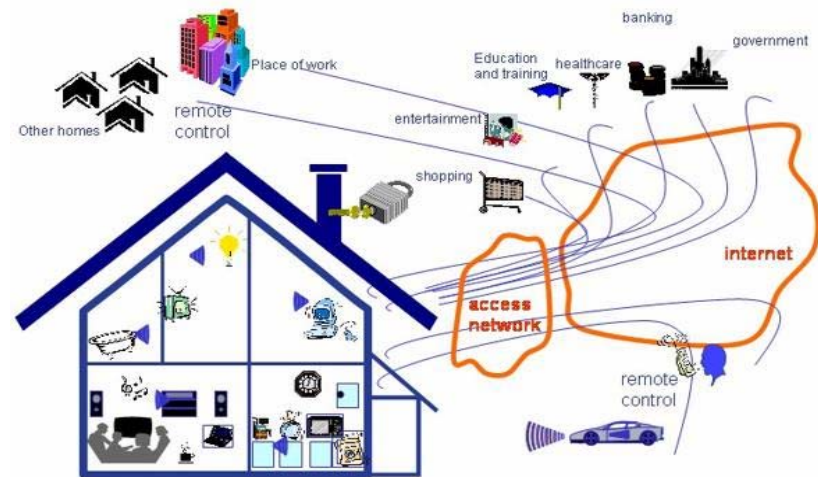


# Ubiquitous Computing (UC)

A revolution in the field of computation, communication and interaction is taking place, and Weiser's vision described in 1991 is becoming a reality.

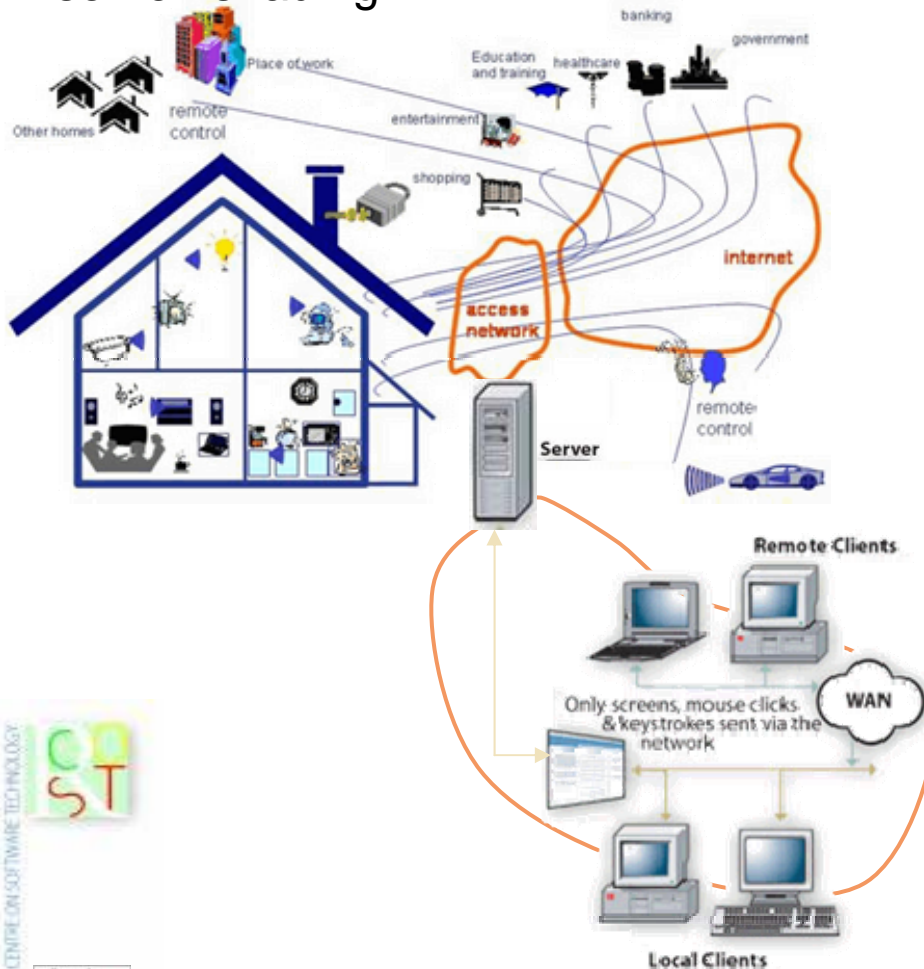
Protocol	Note
Batibus	Widely used in France. <a href="http://www.batibus.com/">http://www.batibus.com/</a>
Bluetooth	In an expansion stage, but broadly supporting short-range wireless technology. <a href="https://www.bluetooth.org/">https://www.bluetooth.org/</a>
EIB	(European electronic interconnection bus) number 1 <a href="http://www.eib.org/">http://www.eib.org/</a>
Ethernet, TCP/IP	Widely used
i-Link	(fire wire systems) <a href="http://www.apple.com/ielink/">http://www.apple.com/ielink/</a>
LON	(Local Operating System) widely expected in 5 years <a href="http://www.lonworks.com/">http://www.lonworks.com/</a>
X-10	Primarily used for home automation <a href="http://www.x10.com/">http://www.x10.com/</a>
Service and discovery-focused standard	Note
HAVi	Home Audio/Video interoperability specification. It defines a set of protocols/APIs that include device abstraction /device control models, an addressing scheme/lookup service, an open execution environment, Plug-n-Play capability (through 1394), and management of isochronous data streams. <a href="http://www.havi.org/">http://www.havi.org/</a>
Jini	It is a middleware layer that resides on top of a Java Virtual Machine and uses Java's Remote Method Invocation (RMI) to make use of a remote device's services. Jini technology has a lookup service with which devices and services register. <a href="http://www.jini.org/">http://www.jini.org/</a>
OSGi	(Open Service Gateway initiative). It is a platform and communication medium independent and supports multiple local network technologies whether wired or wireless. It also supports device access technologies as UPnP and Jini as well as java as an implementation platform. <a href="http://www.osgi.org/">http://www.osgi.org/</a>
UPnP	(Universal Plug and Play). It is an initiative for the discovery and service APIs for a dynamic environment as might be found in the home. <a href="http://www.upnp.org/">http://www.upnp.org/</a>

- the world will consist of ubiquitous systems rather than “*the ubiquitous system*”;
- large, highly dynamic distributed systems, always working and available, will be conceived;
- “*browsing reality*” will be similar to browsing the web.



# NCC towards UC (I)

The widespread interest in the ubiquitous or pervasive systems will give new developments to NCC. In particular, it will influence the notion of *thin client* and consequently, that of NCC middleware. The difference between client and server is fading.



Today OSGi enables the life cycle management of services in a gateway permitting the connectivity between different internal networks (e.g., X-10, CAN bus, EIB) letting hardware devices to interoperate and external networks (e.g., Internet) for controlling and monitoring the devices through different user interfaces (e.g.: Web/HTML, WAP/WML, Java Swing). For enterprise-scale: J2EE.

**When small hardware devices will have the capacity to embed Java Standard Edition, Jini, ServiceUI and Jini/RIO. They may become the foundations for the next wave of NCC/UC:**

- Platform provisioning
- Application Provisioning:
  - start/stop applications.
- Dynamic Service Provisioning: SLA,...

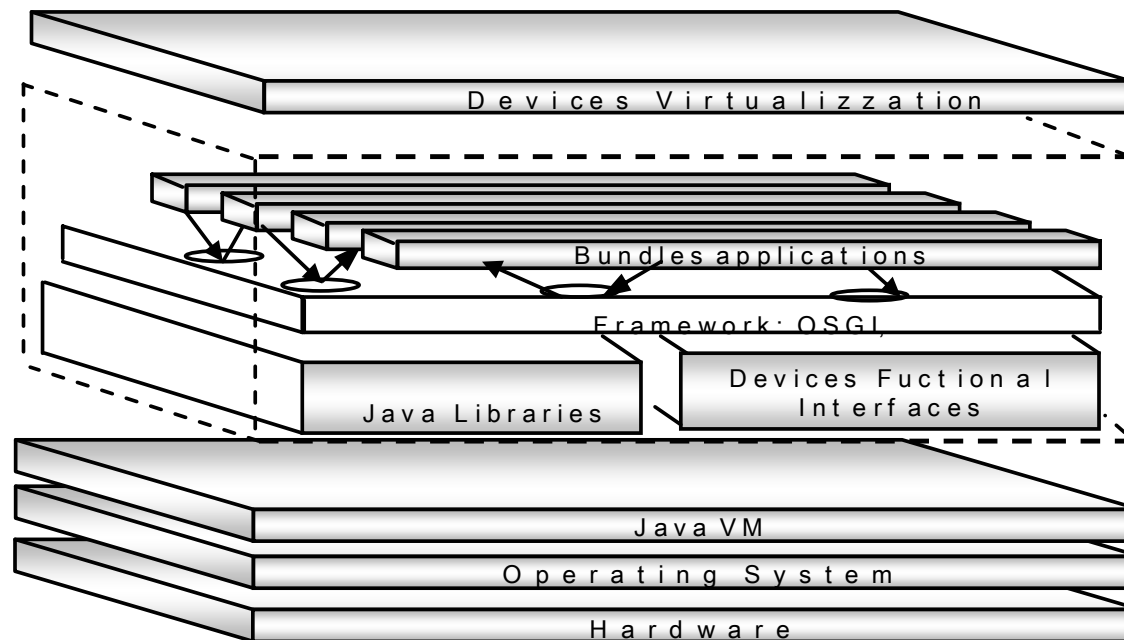




# NCC towards UC (II)

We need to define not only interface between layers (HTML, HTTP, RMI, ...) but also to define standard interfaces between services in order to enable different providers of services to use each other. It will also simplify ubiquitous thin client development by using higher level of devices abstraction and providing an event-based communication mechanism used between ubiquitous thin clients for asynchronous change notification.

We developed an abstraction level named Devices Virtualization layer at a programmatic syntactic level (instead of at the physical link level) in the context of Intelligent Home automation.

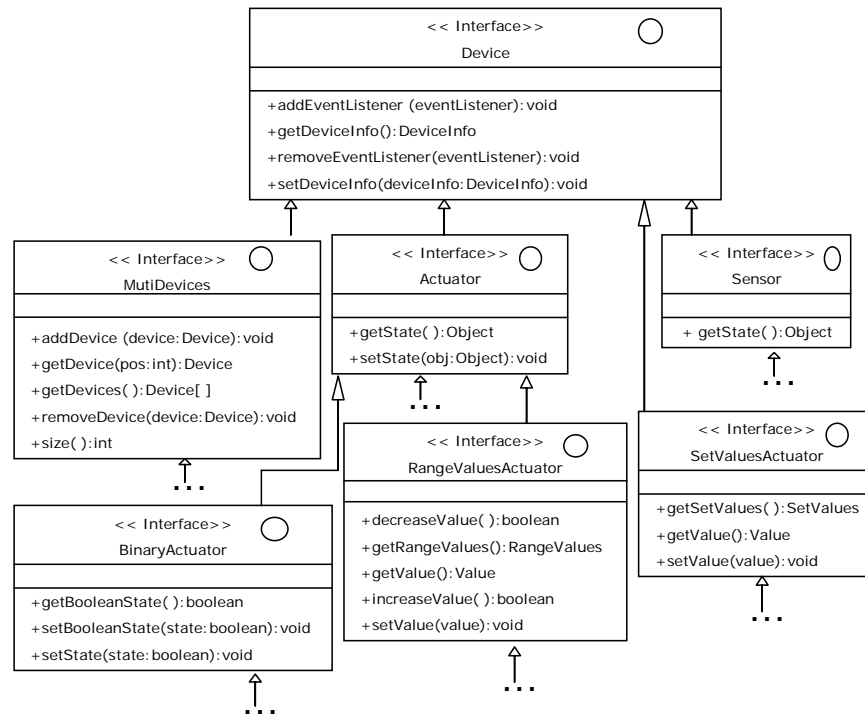


# Devices Virtualization Layer

Two main families of technological products can be identified :

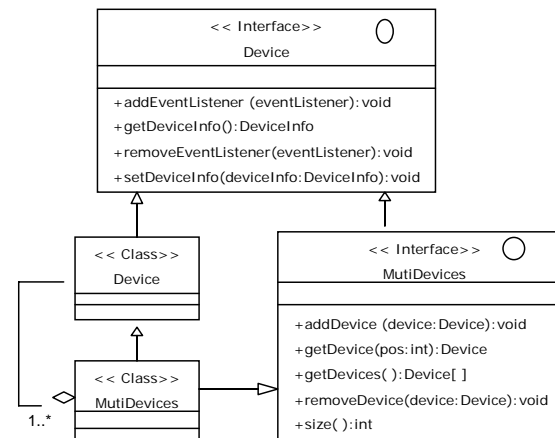
- a) *Sensors*, capturing information from the networked devices and/or the environments, and, then, producing events;
- b) *Actuators*, consuming events and, then triggering actions on the networked devices in the considered environments.

*Sensors and Actuators* can be still specialized in other objects.



Besides the *Sensors* and *Actuators*, complex devices exist in the living environments.

We need as well non-functional aspects: attributes based, ...



# Conclusion

- Each hardware devices should provide all/some of its' services in a remotely accessible way
  - Downloadable User interface (At Content level: HTML, WML, ... and/or at Application level: Java/AWT, Java/Swing, ... => ServiceUI)
  - Programmatic syntactic interface level: standard programmatic syntactic interface or dynamically adaptable programmatic syntactic interface (dynamic proxy generation)
  - Semantic access level ?
- The middleware should include facilities to manage the downloaded code (user interface, security, life-cycle, ...) as well as highly dynamic network services (discovery, lease, ...).



# Open Questions

- Should the thin-client and NCC middleware concepts to change for **integrating ubiquitous computing** ?
- How should the existent technologies change for supporting the **evolution of the thin-client** and NCC middleware concepts ?
- How should traditional interface evolve for allowing common people to **browse the reality** in the most natural way?
- Which instruments should NCC developers offer to common people for **preserving their security and privacy**?
- What may be needed for **reaching standardization** on virtualization of remotely accessible highly dynamic thin clients?

