Middleware for Mobile and Ubiquitous Computing

Ass. Prof. Jean-Yves Tigli – http://www.tigli.fr
at Polytech of Nice Sophia Antipolis University
Currently delegated at INRIA in the team PULSAR

Email : tigli@polytech.unice.fr

In collaboration with S. Lavirotte and G. Rey

IAM01 Course for Polytech'Nice SI5, Master IFI and International Master Ubinet

SCHEDULE OF THE MODULE IAM01

coursee Parts	date	session 1				session 2			
		schedule	type of session	Title	speaker/ teacher	schedule	type of session	Title	speaker/ teacher
Technologies for Mobile and Ubiquitous Middleware	06/10/2009	08h30-10h30	*course*	Les grandes Tendances des Middlewares pour l'Informatique Mobile : Hétérogénéité des dispositifs impose des frameworks aux frameworks manageables extensibles	*J.-Y. Tigli*	10h45-12h45	*course*	Extension des frameworks, Comment ? : exemple sous .Net natif / managé	*J.-Y. Tigli*
	13/10/2009	08h30-10h30	*course*	Extension des frameworks, Comment ? : exemple sous Java, natif / managé : JNI, CNI	*J.-Y. Tigli*	10h45-12h45	*Pratical course*	Extension des frameworks, Comment ? : exemple sous Java, natif / managé : JNI, CNI	*J.-Y. Tigli*
	20/10/2009	08h30-10h30	*course*	Exemple de Framework pour Mobile : J2ME	*M. Riveill*	10h45-12h45	*Pratical course*	Exemple de Framework pour Mobile : J2ME	*M. Riveill*
	27/10/2009	08h30-10h30	*course*	Exemple de Framework pour Mobile : J2ME	*M. Riveill*	10h45-12h45	*course*	Exemple de Framework pour Mobile : J2ME	*M. Riveill*
	03/11/2009	08h30-10h30	*Pratical course*	Exemple de Framework pour Mobile : Android	*G. Rey ou A. Farine*	10h45-12h45	*Pratical course*	Exemple de Framework pour Mobile : Android	*G.Rey ou A. Farine*
	10/11/2009	08h30-10h30	*course*	Exemple de Framework pour Mobile : Compact .Net Framework	*J.-Y. Tigli*	10h45-12h45	*Pratical course*	Exemple de Framework pour Mobile : Compact .Net Framework	*J.-Y. Tigli*
	17/11/2009	08h30-10h30	*course/Pratical course*	Exemple de Framework Manageable Extensible : OSGi	*D. Donsez ou autre intervenant LIG*	10h45-12h45	*course/Pratical course*	Exemple de Framework Manageable Extensible : OSGi	*D. Donsez ou autre intervenant LIG*
	24/11/2009	08h30-9h30	*course*	Extension des frameworks, pourquoi ? : Example, Context-aware extensions for Middleware	*G. Rey*	09h45-12h45	*Pratical course*	Example : Context Toolkit	*N. Ferry*

SCHEDULE OF THE MODULE IAM01

Researches on Mobile and Ubiquitous Middleware	01/12/2009	08h30-10h30	*course*	Trends and requirements in Middleware for Ubiquituous Computing : Dynamic Evolution of the software environment impose Adaptation at run time	*J.-Y. Tigli*	10h45-12h45	*course*	Service oriented Middleware and Dynamic composition	*J.-Y. Tigli*
	08/12/2009	08h30-10h30	*course*	Example : LCA/SLCA models in WComp (WS, UPnP, DPWS)	*S. Lavirotte*	10h45-12h45	*Pratical course*	Example : LCA/SLCA models in WComp (WS, UPnP, DPWS)	*S. Lavirotte*
	15/12/2009	08h30-10h30	*course*	Exemple : LCA/SLCA models in WComp (Composition)	*J.-Y. Tigli*	10h45-12h45	*Pratical course*	Exemple : LCA/SLCA models in WComp (Composition)	*J.-Y. Tigli*
	05/01/2010	08h30-9h30	*course*	Adaptive Middleware at Runtime	*J.-Y. Tigli*	10h45-12h45	*course*	Example : AA model in WComp	*J.-Y. Tigli*
	12/01/2010	08h30-10h30	*Pratical course*	Student Project: Research Bibliography Study / Technology Forecasting	*J.-Y. Tigli*	10h45-12h45	*course/Pratical course*	Student Project: Research Bibliography Study / Technology Forecasting	*J.-Y. Tigli*
	19/01/2010	08h30-10h30	*course/Pratical course*	Student Project: Research Bibliography Study / Technology Forecasting	*S. Lavirotte*	10h45-12h45	*course/Pratical course*	Student Project: Research Bibliography Study / Technology Forecasting	*S. Lavirotte*
	26/01/2010	08h30-10h30	*course/Pratical course*	Student Project: Research Bibliography Study / Technology Forecasting	*G. Rey*	10h45-12h45	*course/Pratical course*	Student Project: Research Bibliography Study / Technology Forecasting	*G. Rey*
	02/02/2010	**Exam**							

Université Nice SOPHIA ANTIPOLIS

OUTLINE OF THE FIRST PART

- Early challenges in Middleware for Ubiquitous Computing
- First software approaches to deal with heterogeneity
- Software trends to deal with heterogeneity : Extensible Frameworks
  - Ex. MEF Managed Extensibility Framework
  - Ex. OSGi

Early challenges in Middleware for Ubiquitous Computing

What Ubiquitous Computing does mean ?

What middleware does mean ?

First challenges for middleware for ubiquitous computing …

Software trends to deal with heterogeneity

…

What Ubiquitous Computing does mean ?

What does Ubiquitous Computing mean ?

- [Mark Weiser 1991]

« Silicon-based information technology, is far from having become part of the environment. »

Scientific American,
Vol. 265 N.9, pp. 66-75, 1991

Ubiquitous applications used everyday life connected objets and devices

FIRST UBIQUITOUS APPLICATIONS

- Smart Objets and Devices are well-known at Design time, Embedding for smart control
  - Embedded systems for cars, airplanes, etc
- First Ubiquitous Application are generally Ad-Hoc applications without middleware, Creating new computing devices
  - Hi-tech, silicon-based gadgetry, e.g. PDAs, cell phones, mp3 players, active displays

FIRST REQUIREMENTS

- First requirements :
  - System requirement : Ubiquitous Applications applications are continuously interacting with a real world
  - Design requirement : Smart objects and devices must be able to communicate spontaneous information from the environment to the application
  - Software requirement : Software application must be event-driven

New Constraints for ubiquitous computing : Heterogeneity of devices

- Technological Heterogeneity of smart objects and devices
  - Numerous software and network technologies

- But also Semantic Heterogeneity
  - Various Smart Objects and Devices (sensors, mobile phones, …, coffee machine, mug …)
  - Variation of capabilities between them (ex. from J2ME to JSRs in mobile phones)

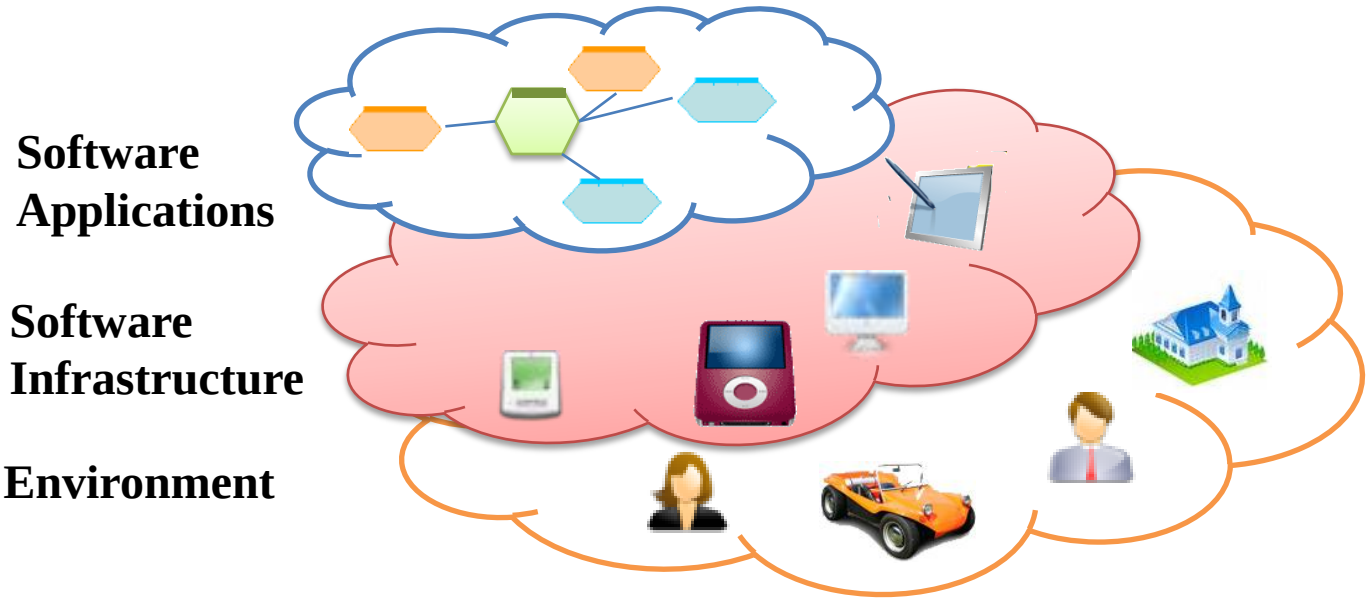
New Constraints for ubiquitous computing : Mobility

- WSI user-centered reference Model
  - Spheres of interaction of devices, from Personal Area Network to World Wide Web

S. Arbanowski, M. Lipka, K. Mössner, K. Ott, R. Pabst, P. Pulli, A. Schieder, M. A. Uusitalo. The **WSI Reference Model** for the Wireless World. *Proceedings of IST Mobile Summit 2003*.

- Users and Devices are Mobile

Main Ubiquitous Computing Characteristics



- Three main characteristics are :
  - Use embedded devices in a real environment
  - Deal with Multiple Heterogeneous Devices
  - Deal with Highly Dynamic variation at Runtime

Requirements for software composition between smart objects and devices

- Main requirements for composition are :
  - *Interating with Real World* => Event based interaction in the composition
  - *Heterogeneous Devices* => Discover at runtime, new smart objects and devices
  - *Mobility* => Deal with dynamic appareance and dispareance of smart objects and devices
  - *Mobility* => Deal with dynamic composition (at runtime)
  - *Mobility* => Distribution must explicit to deal with the evolution of the infrastructure (we distinguish local and distributed composition)

MULTI-DOMAIN ADAPTATION AS OPEN ISSUE

- Ubiquitous Applications continuously adapt at runtime, application requirements to changing computing environment (due to mobility) in multiple domains :
  - HMI,
  - Power,
  - Network bandwidth,
  - Devices availability, …

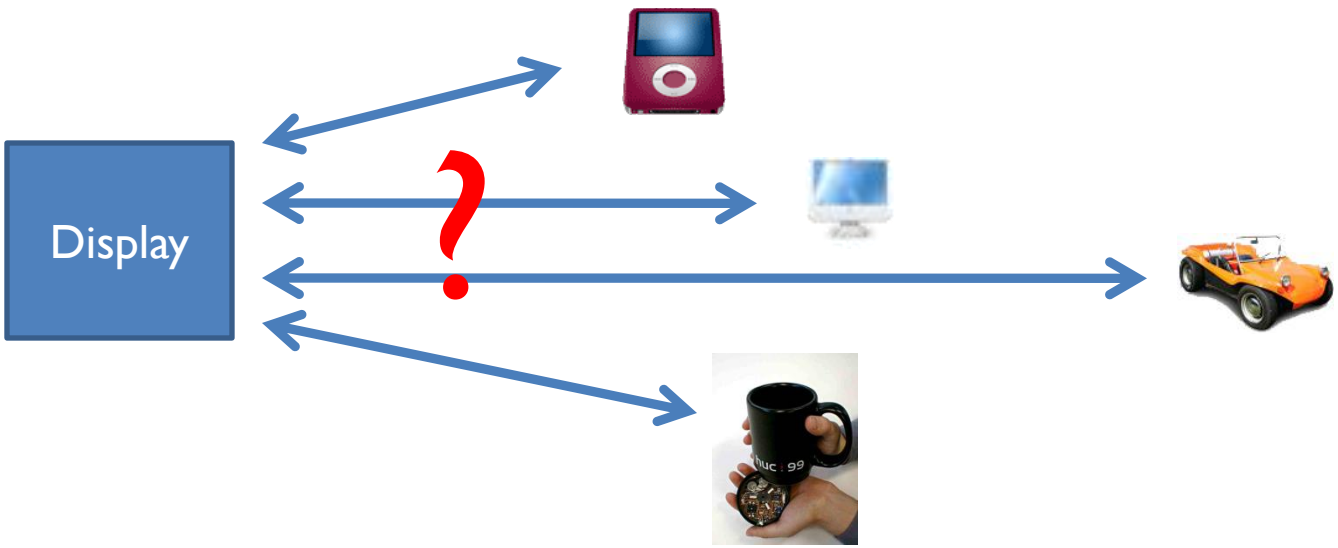
REACTIVE ADAPTATION AS OPEN ISSUE

- Reactive adaptation is defined as the ability for the Ubiquitous applications to perceive the environment and adapt to changes in that environment in a timely fashion.

- Ubiquitous Applications must provide reactive adaptation mecanism to changing operational environment.

SEMANTIC ADAPTATION AS OPEN ISSUE

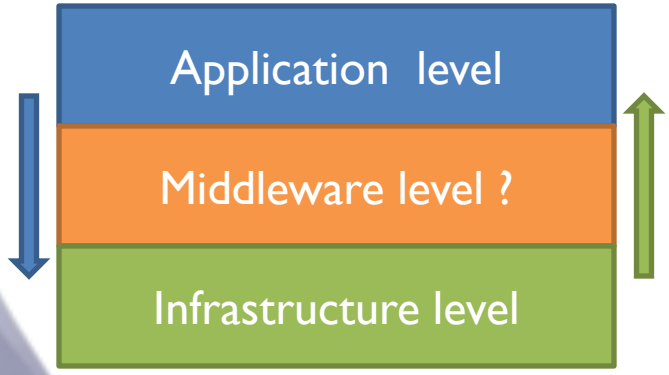
- Application requirements must match at run-time the current operational environment.



Can match with ?

New challenges and open issues in adaptation

- Ubiquitous Computing applications are continuously interacting with a real world, partly unknown at design time and, always changing at runtime in uncountable manner
- We witness to a kind of inversion in the classical software methodology where the software applications levels are much more stable and stationary than the software infrastructure level.



WHAT MIDDLEWARE DOES MEAN ?

From traditional Middlewares for distributed applications to new Middlewares trends…

DEFINITIONS

- *... middleware must allow the interoperability of the applications that are based on it.* (Laurent Kott, General Delegate to Technology Transfers at INRIA. (2002))

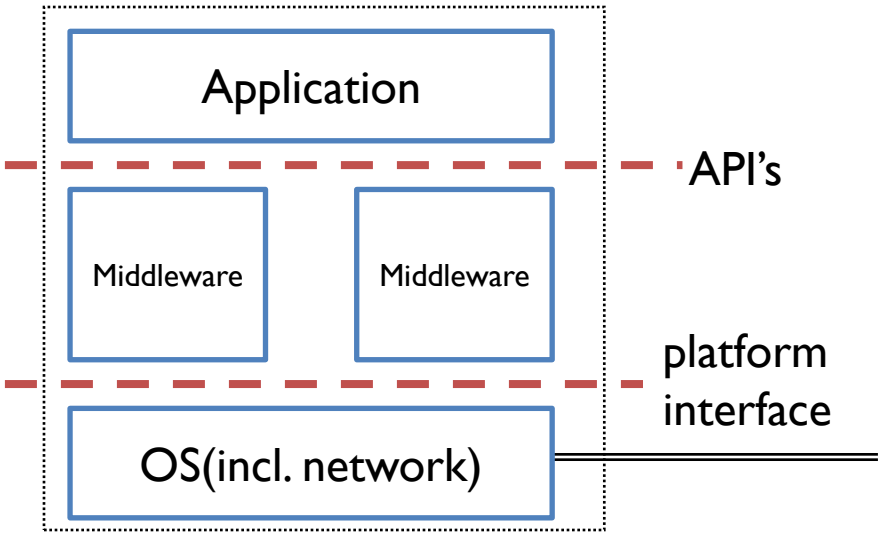
- *the intersection of the stuff that network engineers don't want to do with the stuff that applications developers don't want to do.* (Kenneth J. Klingenstein ('99))

DEFINITIONS

- *Middleware can be viewed as a reusable, expandable set of services and functions that are commonly needed by many applications to function well in a networked environment.* (NGI workshop, '97)
- *...middleware is a general term for any programming that serves to "glue together" or mediate between two separate and usually already existing programs.* (...)

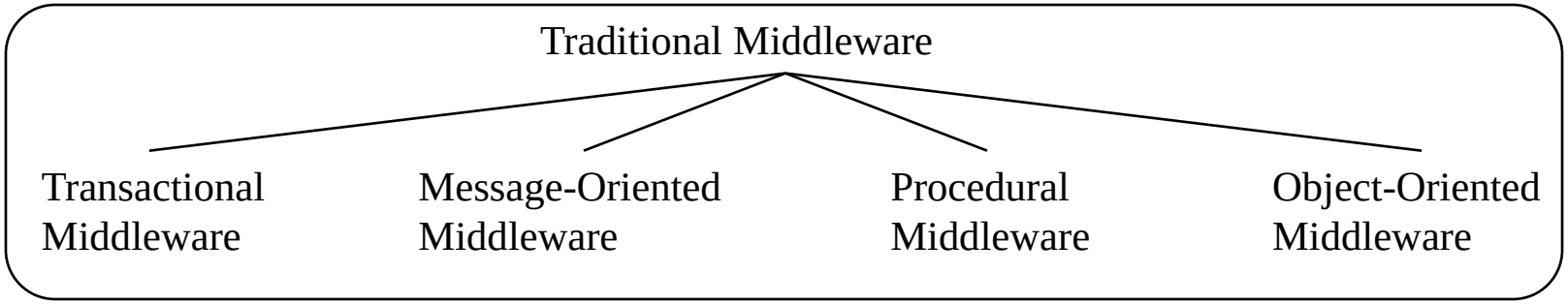
Traditional Middleware for distributed systems

- Networking
- Interoperability

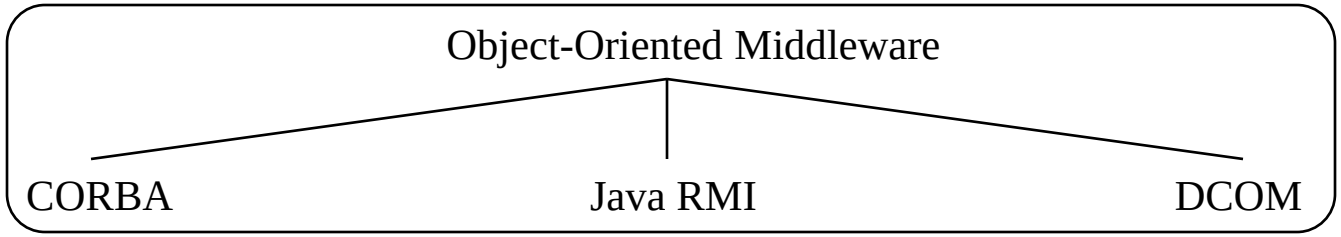


Middleware Classification by Emmerich

- Message Oriented Middleware (MOM)
- Transaction Processing Monitors (TPMON)
- Remote Procedure Calls (RPC)



- Object Oriented Middleware (ORB, ...)



Message Oriented Middleware (MOM)

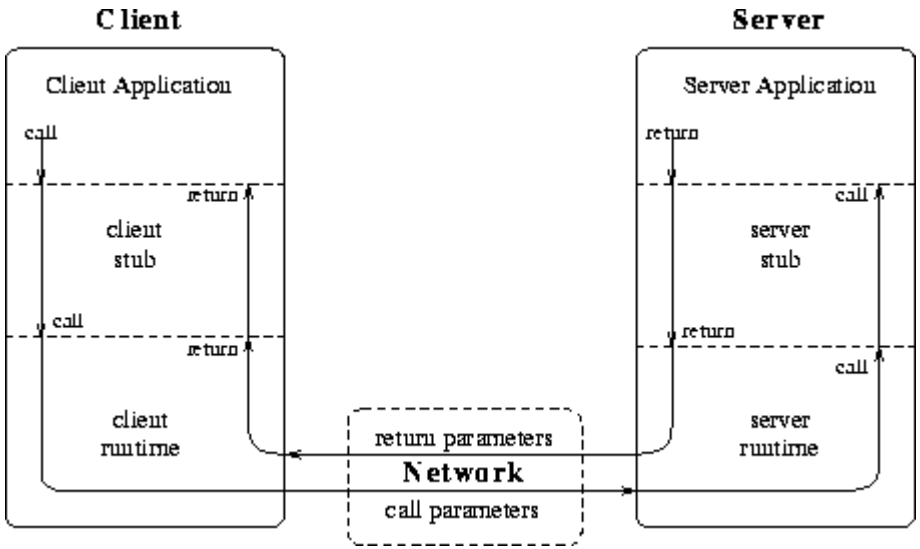
- Basic service: message transportation
  - point-to-point, channels
  - mostly asynchronous .... message queues
  - publish/subscribe: notification
- Quality in delivery types
  - *must* or *should* arrive (e.g., JMS)
  - *durable*: retain message across connection-loss
- Message properties
  - priority
  - order
  - expiration time

Transaction Processing Monitors

- Transaction: series of communications satisfying the ACID properties
  - atomic transaction is taken or not taken
  - consistent system invariants are maintained
  - isolation transient states not observable
  - durable transaction is persistent
- Associated strongly with distributed data(base) access
  - transactions implement business logic
- Can be built on top of e.g., MOM

Remote Procedure Call

- Maintain local procedure semantics



- Generate server and client stub from IDL
- Synchronous
- No real middleware (no API)

Object Oriented Middleware

- Object oriented: data+method remote
  - DCOM (on top of COM, DCE), RMI (JAVA based), Corba (Object Request Brokers)

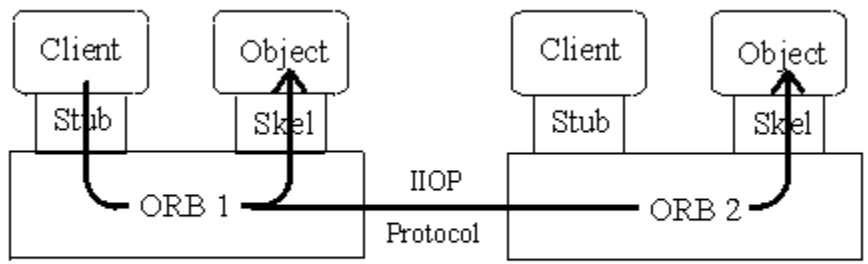


Figure 2: Interoperability uses ORB-to-ORB communication

- ORB as an intermediate ("bus")
  - dynamically provide server-object references
  - variety of other tasks and services

And… Event based Middleware

- Notion of an Event:
  - Asynchronous occurrence in time
  - Contains data that describes the occurrence
  - Can be implemented as message
- Event sources publish events; Event sinks subscribe to events with a filter expression (subscription)

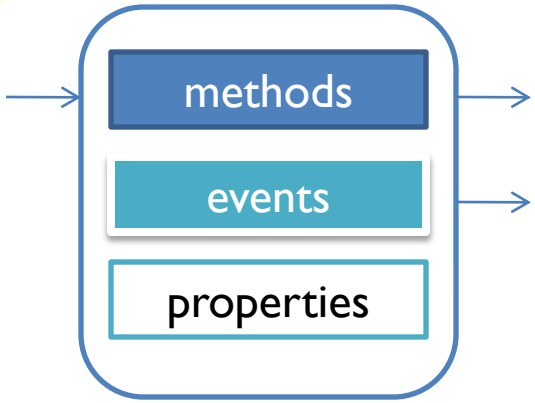
- Properties:
  - Asynchronous notification of events
  - Publishers/Subscribers are decoupled in space and time
  - Many-to-many Communication
  - Anonymity, fault-tolerance, …?

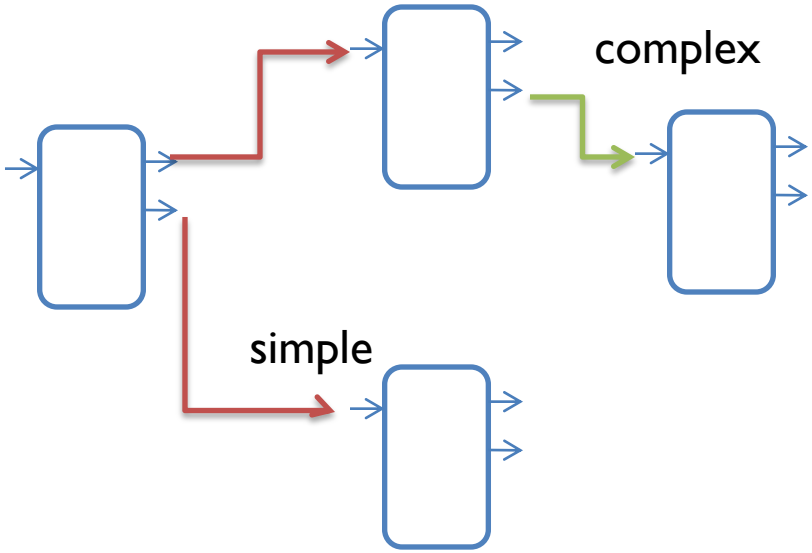
EXAMPLE : LCA/WCOMP MIDDLEWARE

- Main requirements :
  - Composition must be event based
  - Composition is local (like OpenCom)
  - At runtime ….
- Solution :
  - Event based Local Composition : LCA (Lightweight Component Model) for each application execution node.
- Principles
  - LightWeight Components Approach :
    - Like OpenCom, JavaBeans, PicoContainer
  - A container dynamically manage the assembly of components
  - Event-based interaction between components
  - Blackbox LightWeight Components

II.2 LCA COMPONENTS, PORTS AND CONNECTORS

LCA components





Connectors

Simple Event based Connector

C1.Event (param) → C2.Method (param)

Complex Event based Connector

C1.Event (param) → C2.Method (C1.GetAProperty())

Other Definitions

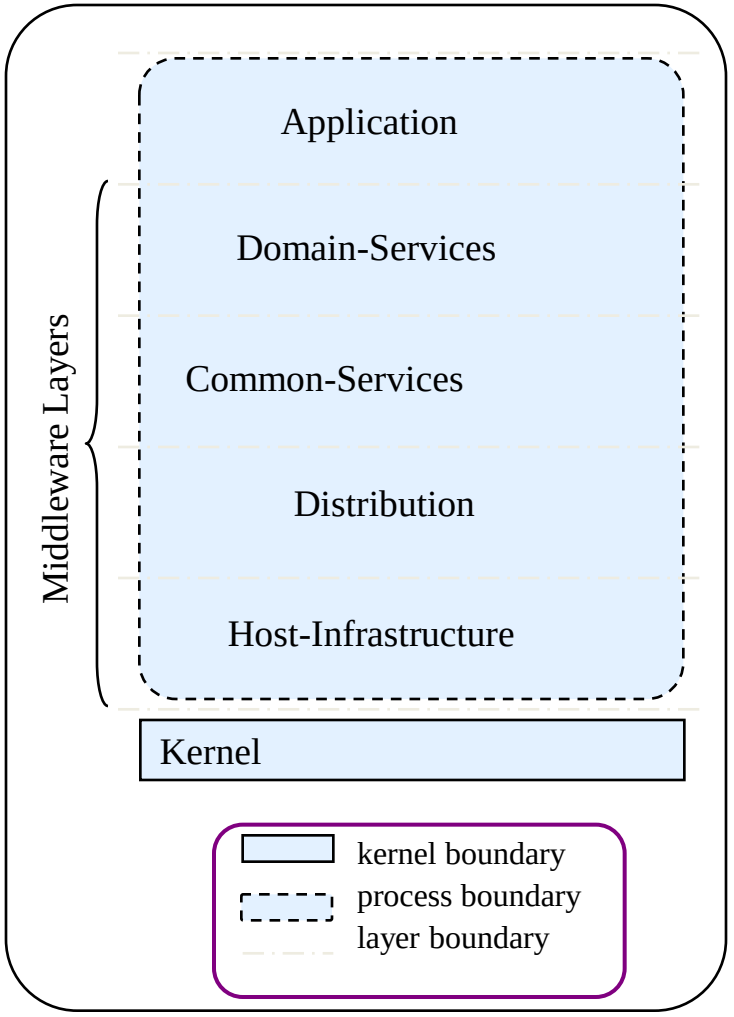
- *... a customizable set of components which can be tailored to the needs of an application.* (Middleware 2001)
- *Software that provides a link between disparate applications.*

  (Computer user dictionary; Lycos tech dictionary)

=> Intermediary software layers to facilitate *resources* management.

Middleware Layers

- Schmidt decomposed middleware into four layers:
  - Domain-Services
    - Tailored to a specific class of distributed applications
  - Common-Services
    - Functionality such as fault tolerance, security, load balancing and transactions
  - Distribution
    - Programming-language abstraction
  - Host-Infrastructure
    - Platform-abstraction



1. Douglas C. Schmidt. Middleware for real-time and embedded systems. *Communications of the ACM*, 45(6), June 2002.

HOST-INFRASTRUCTURE LAYER

- The host-infrastructure layer resides directly atop the operating system kernel and provides a higher-level API than the operating system API that hides the heterogeneity of hardware platforms, operating systems and, to some extent, network protocols.

- Example : JVM / J2SE
- Example : CLR / .Net Framework

Sadjadi, M. and McKinley, P.K, "*A survey of adaptive middleware*". *Technical Report MSU-CSE-03-35, 2003*.

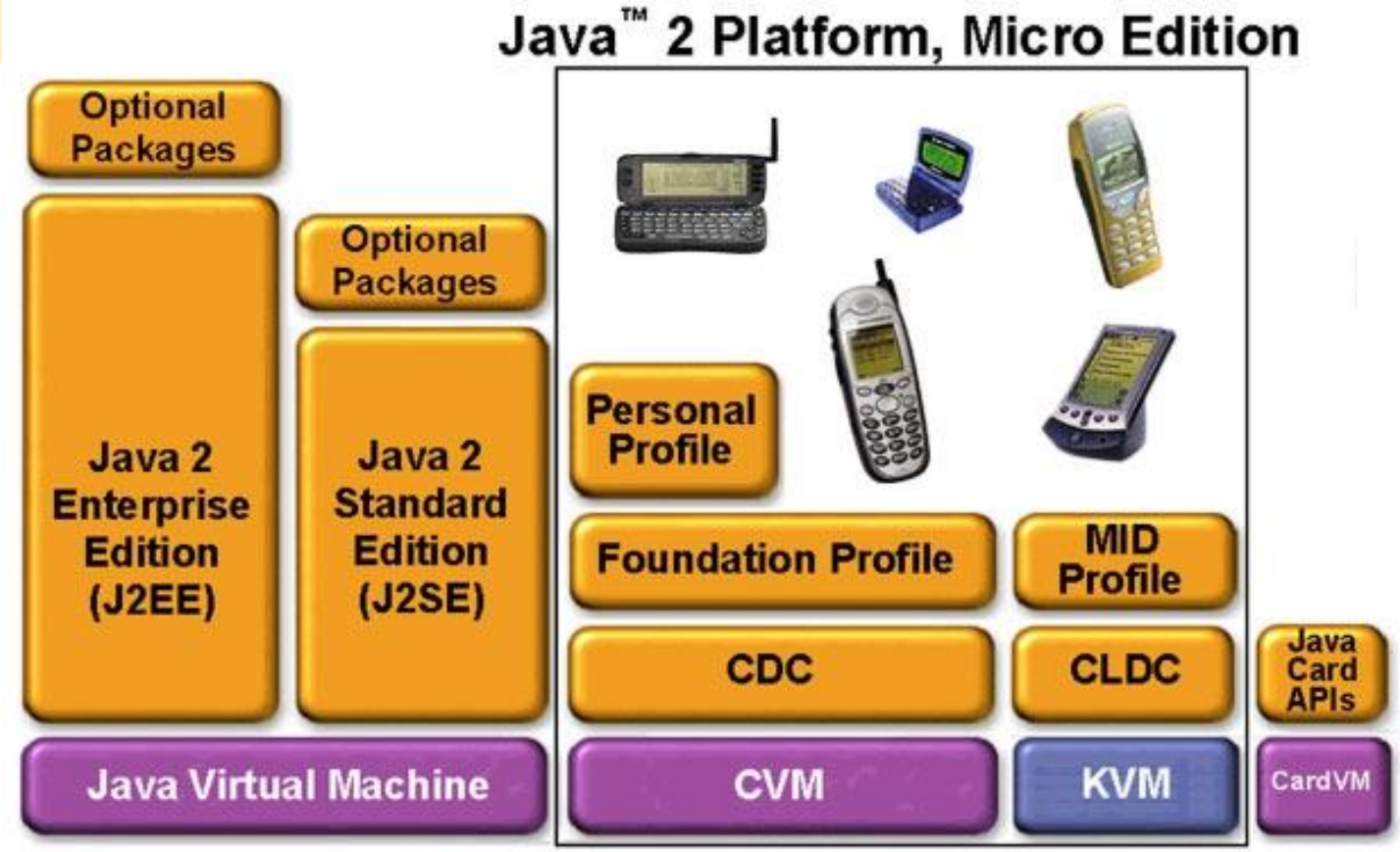
FIRST TROUBLES WITH HETEROGENEITY …

- DEMO

<< With or without touch screen ? >>

.NET FRAMEWORKS

JAVA EDITIONS



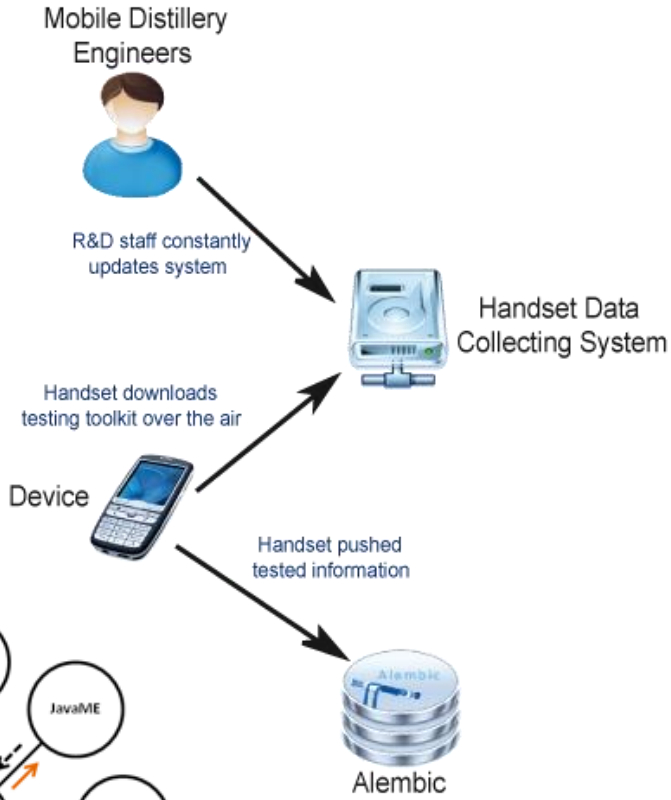
- And more and more JSR …

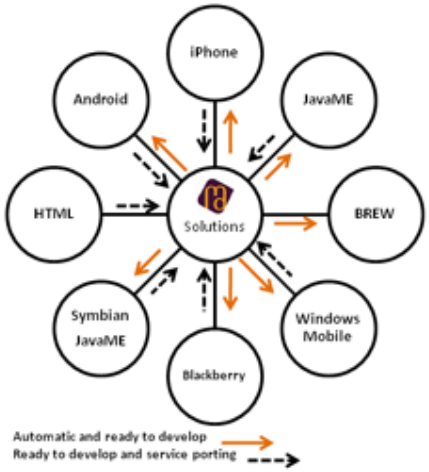
Java Community Process

Université Nice SOPHIA ANTIPOLIS

EXPLOSION OF THE STANDARDS …

- Test your device …
- https://www.alembic-database.com/
- New Software Business
- http://www.mobile-distillery.com





Next Technical Courses

- First software approaches to deal with heterogeneity
  - Frameworks and Native interfaces
  - (CLR P/Invoke, JVM / JNI)
  - Standards Frameworks for Mobile Devices (J2ME, Android, .Net Compact Frameworks)
- Software trends to deal with heterogeneity : Extensible Frameworks
  - Ex. MEF Managed Extensibility Framework
    Ex. OSGi

Next Technical Courses

- TC 1 : Introduction to CLR P /Invoke
- TC 2 : JVM structure and JNI (Java Native Interface)
- TC 3 : OSGi
- Free TC 4 : MEF (Managed Extensible Frameworks)

REFERENCES

1. Douglas C. Schmidt. Middleware for real-time and embedded systems. *Communications of the ACM*, 45(6), June 2002.
2. Sadjadi, M. and McKinley, P.K, "*A survey of adaptive middleware". Technical Report MSU-CSE-03-35, 2003.*
3. *Gordon Blair, "From Mobility to Ubiquity and Beyond: Challenges to Middleware", Keynotes ACM Mobility'09, Sophia Antipolis*
4. *Mark Weiser. "The Computer for the 21th Century." Scientific American, September 1991.*
5. *Mark Weiser. "Some computer science issues in ubiquitous computing." Communications of the ACM, 36(7):75-85, July 1993.*
6. *Mark Weiser, John S. Brown. "The Coming Age of Calm Technology." 1996.*
7. *M. Satyanarayanan. "Fundamental Challenges in Mobile Computing." Fifteenth ACM Symposium on Principles of Distributed Computing, May 1996.*
8. *M. Satyanarayanan. "Pervasive Computing: Vision and Challenges." IEEE Personal Communications, August, 2001.*
9. *Vincent Hourdin, Jean-Yves Tigli, Stéphane Lavirotte, Gaëtan Rey, Michel Riveill, "SLCA, Composite Services for Ubiquitous Computing", in International Conference on Mobile Technology, Applications and Systems, Sep 2008.*

Université Nice SOPHIA ANTIPOLIS

Appendix : Free Technical Course 4 MEF : Managed Extensibility Framework

- http://sebastien.warin.fr/2009/02/21/413-net-4-introduction-mef-managed-extensibility-framework/
- http://www.dotnet-news.com/lien.aspx?ID=42206
- http://mef.codeplex.com/