Corso di Laurea in INFORMATICA Magistrale

Interazione Uomo-Macchina II

Modulo A

a.a. 2009-2010

.

III

Ambienti per EUD, SSW, esempi

Questi lucidi sono stati preparati da Maria Francesca Costabile, Università degli Studi di Bari, per uso didattico. Essi contengono materiale originale di proprietà dell'Università degli Studi di Bari e/o figure di proprietà di altri autori, società e organizzazioni di cui e' riportato il riferimento. Tutto o parte del materiale può essere fotocopiato per uso personale o didattico ma non può essere distribuito per uso commerciale. Qualunque altro uso richiede una specifica autorizzazione da parte dell'Università degli Studi di Bari e degli altri autori coinvolti.

Toward End-User Development

- ***Users*** are people interacting with software systems
  Even software engineers are users of tools and environments they use for their work (e.g. CASE tools)
- ***End users*** are people not experts in computer science nor willing to be, who use computer systems for their daily activities, for work as well as for entertainment or other
- "Using the system changes the users, and as they change they will use the system in new ways" [Nielsen 1993]
- Interactive systems must be designed to ***evolve*** as needs of ***evolving*** end users might require
- ***Co-evolution*** of end user and system [Costabile et al, AVI 2006]
- Allowing end-users to personalize and evolve at run time their own software environments and tools ➔ ***End User Development***

End-User Development

- "End User Development is a set of activities or techniques that allow users of software systems, who are acting as non-professional software developers, at some point to create or modify a software artefact "

[EUD-Net 2002-2004]

- EUD means the active participation of end users in the software development and evolution processes
- New challenges for system design and implementation
- The need of EUD is not a luxory but a necessity: computer systems modeling some particular "world" are never complete
  - the world changes and new requirements emerge
  - Skilled domain professionals change their work practices over time

[Fischer 2009]

- Systems are in a "perpetual beta"

End-User Developments activities

NO: activities that allow users to **choose** among alternative behaviours (or presentations or interaction mechanisms) **already available** in the application

- e.g., enter parameters

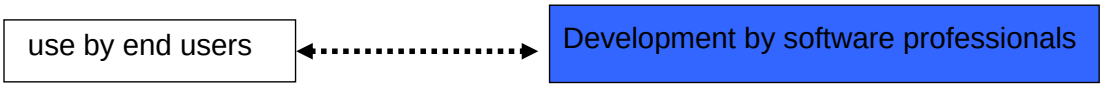
YES: activities that imply some **modification** through any programming paradigm, thus creating or modifying a software artefact

End users' need…

- … powerful and flexible software environments
  - Customized to their culture, skills, needs, age, training, …
  - Tailorable
  - Suitable to manage the user view of the work activity
  - Exploiting concrete, situated, notations: based on icons, symbols, and words that resemble and schematize the tools and the entities used in the working environment
  - Allowing users to personalize and evolve (we say: "to shape") their own software environments ➔ End User Development

End users in sw development

Traditional development



Participatory development

EUD & Meta-design

Meta-design

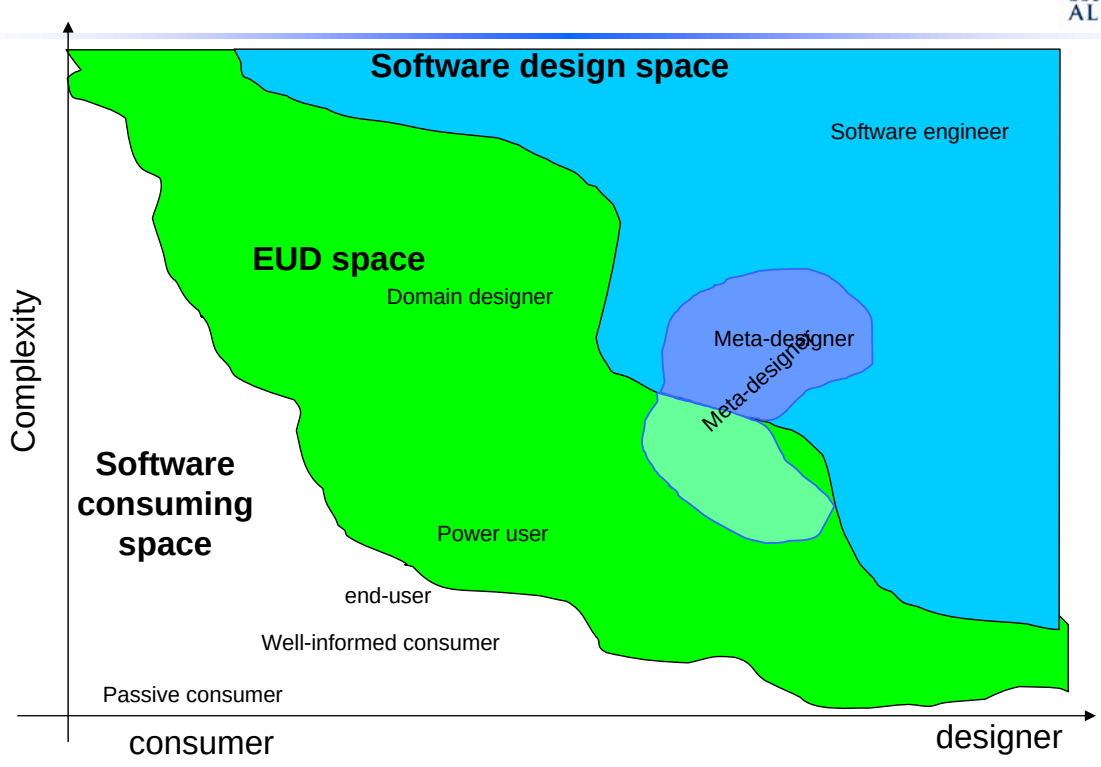
- Design for designers
- It creates open systems at design time that can be modified by their end users acting as co-designers at use time

[Fischer and Giaccardi 2006]

- A design paradigm that includes end users as designers and provides all stakeholders in the team with suitable languages and tools to foster their personal and common reasoning about the development of interactive software systems that support end users' work.

[Costabile et al 2007]

Consumers and designers spaces



Adapted from: Fischer, G., Piccinno, A., Ye, Y.: The Ecology of Participants in Co-Evolving Socio-Technical Environments. In: Forbrig, P., Paternò, F. (eds.) Engineering Interactive Systems, LNCS 5247, pp. 279-286. (2008)

Designing systems that satisfy end users

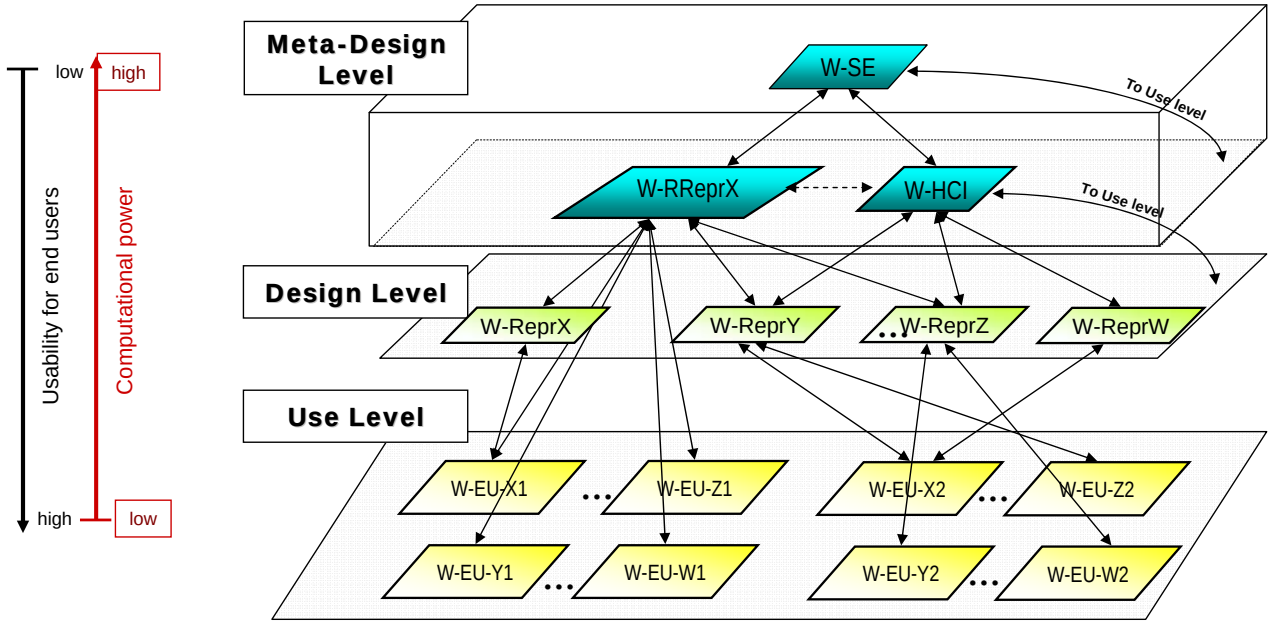
- Some basic principles:

1. In an application domain, there are **different communities of end users**
2. End users need software environments in which they find **all and only the tools** necessary to perform their activities, without being overwhelmed by unnecessary tools and information
3. Systems must be designed for **co-evolution of users and systems**

Software Shaping Workshops

- A useful metaphor for conceptual design: ***artisan workshop***
- A ***Software Shaping Workshop (SSW)*** is a software environment devoted to a specific community of end users
- It is organized as a virtual workshop
  - end users find all and only the tools they need
  - end users carry out their activities and adapt environment and tools using high level visual languages tailored to their needs
- ***Workshop*** as artisan workroom, not as people meeting
- An interactive system to support the work practice in a given application domain
  - cannot provide the same interaction environment to all end users
  - must be designed as a ***set of software shaping workshops***, each specific for a community of end users (SSW methodology)
- In some SSWs, end users perform EUD activities

Interactive system as SSW network

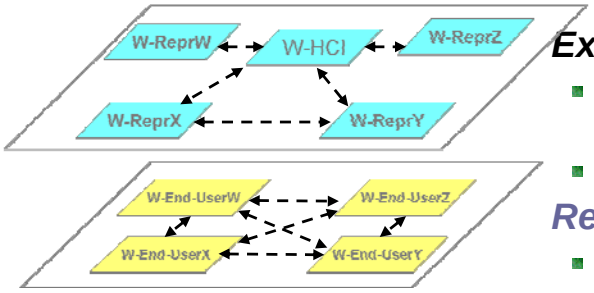


Each SSW provides only the tools to perform the desired activities

SSWs permit to design and evolve a system through collaborative negotiations

- The negotiation is based on the exchange along the SSW network of two types of messages:
  - annotations about these workshops
  - executable specifications of workshops (XML-based documents)
- A stakeholder designs or updates a workshop (e.g. head phisician) by using a domain specific language. His actions modify the executable specification that, when interpreted by the browser, generates the new workshop

Communication paths in the network



Exchange paths:

- the paths along which the exchanges of data and programs occur
- among the workshops at the same level

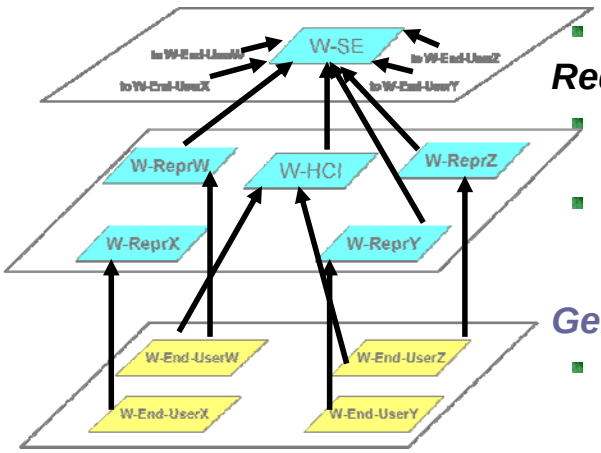
Request paths:

- concerned with the communications going from low levels to higher levels
- trigger the co-evolution process, carrying on the feedback from end users (requests for workshop modification or extension)

Generation paths:

- represent the activity of using workshops at a high level to generate, modify or extend workshops to be used at the lower level
- new or evolved workshops are made available to lower levels along such generation paths

Communication paths in the network



Exchange paths:

- the paths along which the exchanges of data and programs occur
- among the workshops at the same level

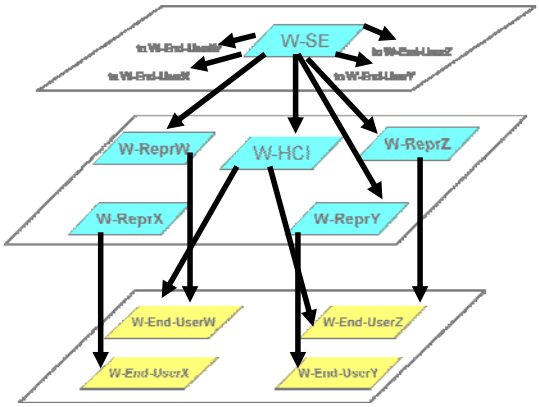
Request paths:

- concerned with the communications going from low levels to higher levels
- trigger the co-evolution process, carrying on the feedback from end users (requests for workshop modification or extension)

Generation paths:

- represent the activity of using workshops at a high level to generate, modify or extend workshops to be used at the lower level
- new or evolved workshops are made available to lower levels along such generation paths

Communication paths in the network



Exchange paths:

- the paths along which the exchanges of data and programs occur
- among the workshops at the same level

Request paths:

- concerned with the communications going from low levels to higher levels
- trigger the co-evolution process, carrying on the feedback from end users (requests for workshop modification or extension)

Generation paths:

- represent the activity of using system workshops at a high level to generate, modify or extend workshops to be used at the lower level
- new or evolved workshops are made available to lower levels along such generation paths

Future scenario for Software

- Recent projects describe a future scenario for software systems
  - Software will increasingly be part of integrated, heterogeneous, and continuously developing infrastructures
  - End users will actively change the tools they work with and orchestrate different services as building blocks to provide the functionality they need
  - In cases where the required functionality cannot be obtained by simple tailoring, they will ***communicate*** and ***cooperate*** with local and other professional designers who in turn might call upon a network of peers or developers of the base products and frameworks used
- The described SSW approach already supports this scenario

Interfacce utente flessibili e personalizzabili

- Le **interfacce utente flessibili** e personalizzabili danno agli utenti la possibilità di configurare la presentazione e le funzionalità del sistema che usano senza dover necessariamente cambiare l'applicazione complessiva

Esempio di interfaccia flessibile: Google, Home page personalizzabile

Case study in the medical domain: designing the Electronic Patient Record (EPR)

- Several projects on EPR
- Many hospital in Italy do not adopt EPR
- We conducted a field study on the use of paper patient record at the “Giovanni XXIII” Children Hospital of Bari, Italy
- Patient record is *many-sided*
  - it is a document to be read and understood by various and very different actors: physicians, nurses, patients’ relatives, the family doctor, etc.,
  - it must have the ability to speak different “voices”, i.e., to convey different meanings according to the actors using it

From the field study

- Various operators use patient records in different ways and to accomplish different tasks
- Specific patient records for each ward, even in the same hospital
- Example:
  - in a children neurological ward, information about newborn feeding must be available
  - in an adult neurological ward, information about alcohol assumption is required
  - In a surgery department, a lot of specific data collected by different specialits (cardiologists, pneumologists, ..) to indicate if the patient can undergo a surgery are required
- Operators are reluctant to accept a common unified format, they want to adapt the patient record to their specific needs

EPR design

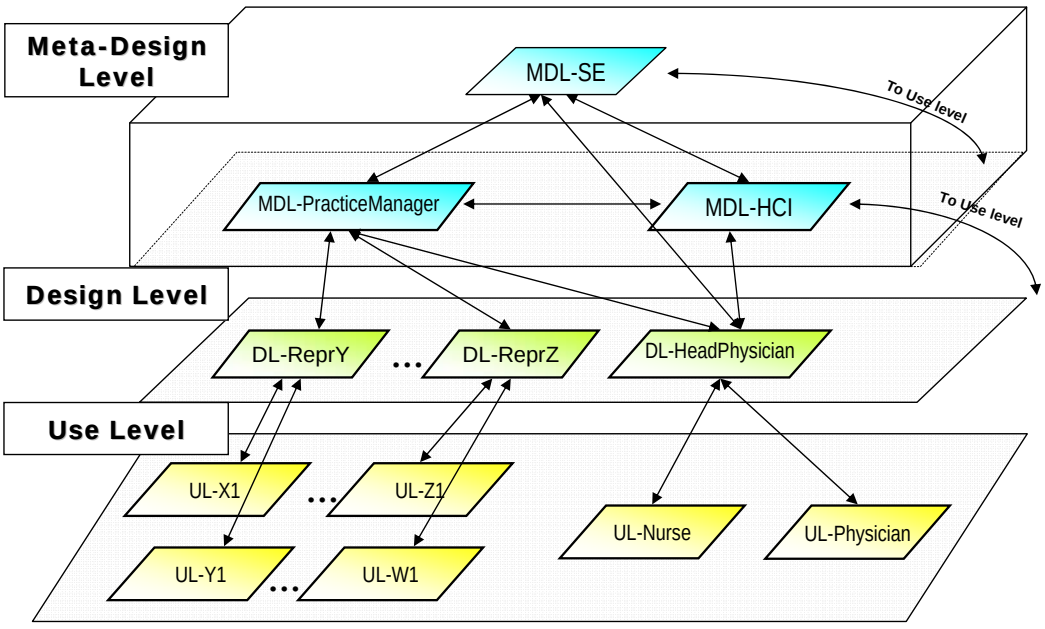
- Patient records can be seen as being composed by modules for collecting patient data
- The patient records used in different wards assemble a subset of modules in different ways, customized to the need of the specific ward

➔

Our approach

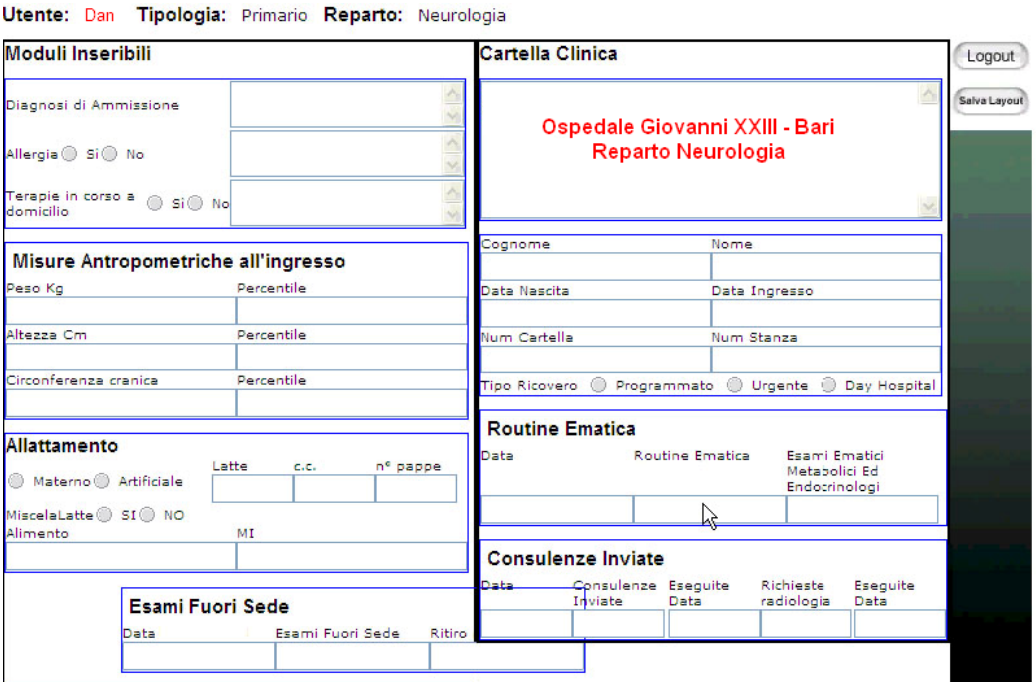
- to identify the data modules that have to be managed in the whole hospital
- let each head physician to design the EPR for her/his ward by composing the ERP through direct manipulation of such modules
- physicians, nurses, … will use this ERP

The SSW network in the case study



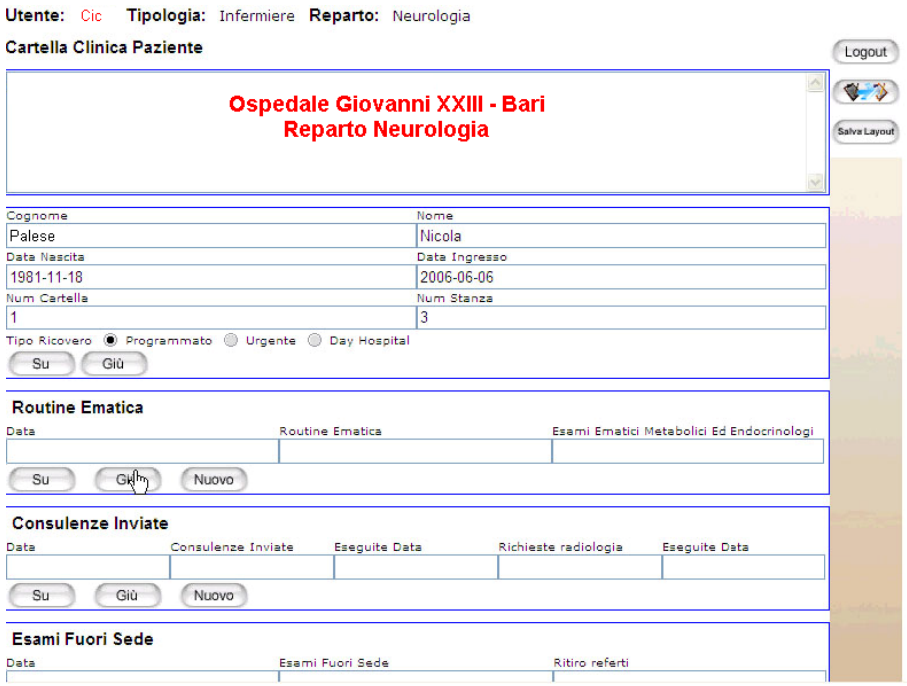
Workshop for Head Physician

W-RapprPrimario is used by each head physician to create and update the specific application workshops “W-End-Infermiere”

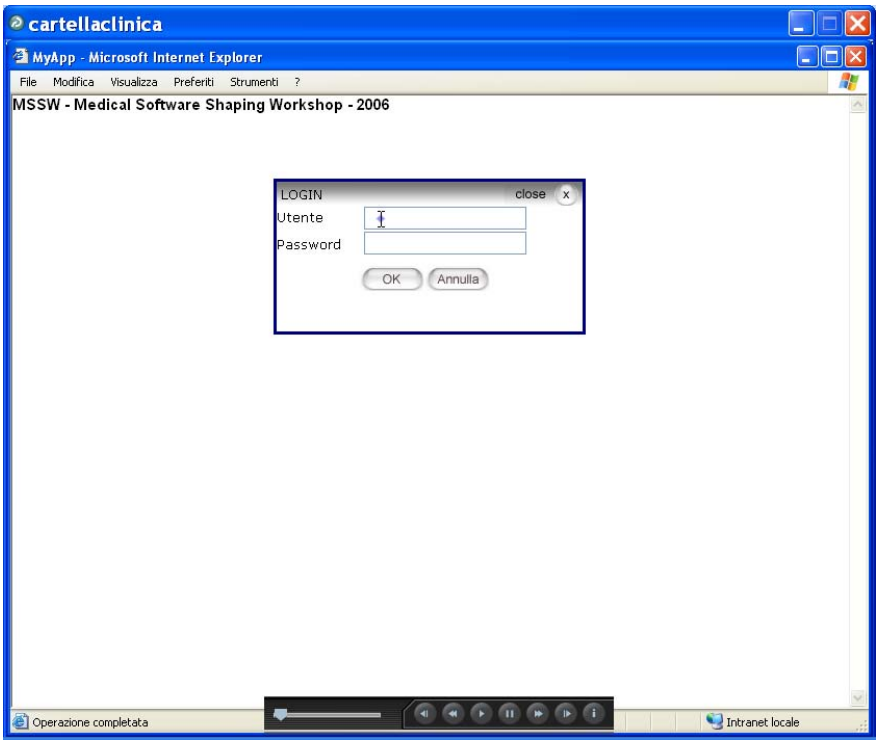


Workshop for Nurses

- **W-End-Infermiere** is used by nurses to daily fill in the measurements on the patient (and required in the patient record)



video



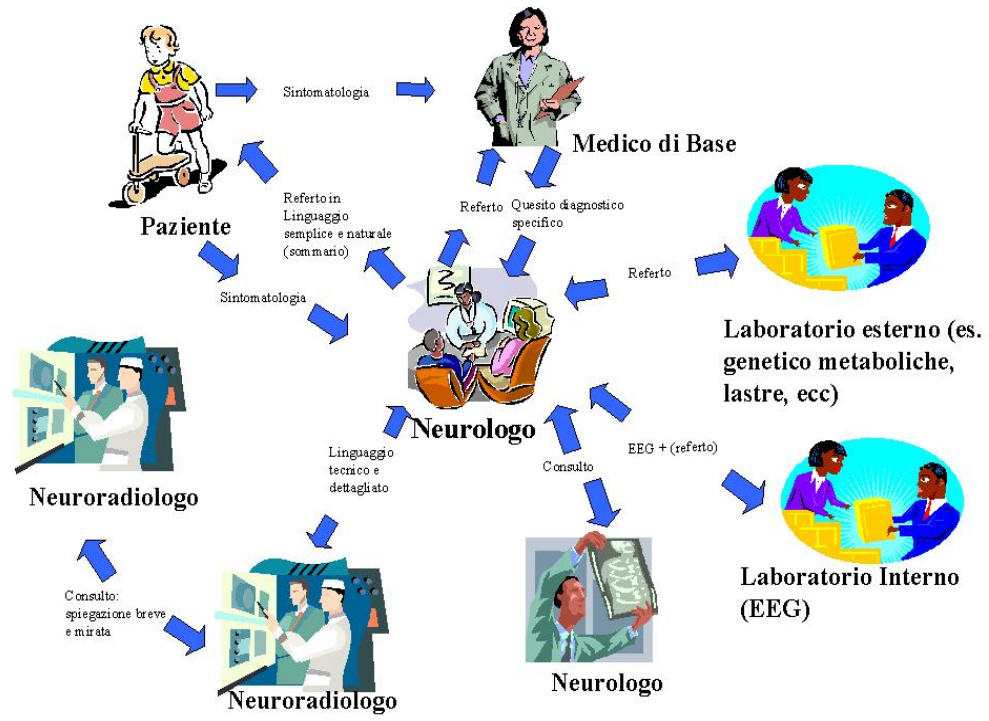
Another example in the medical domain

- Physicians of two categories collaborate to reach a diagnosis:
  - neurologists
  - Neuro-radiologists

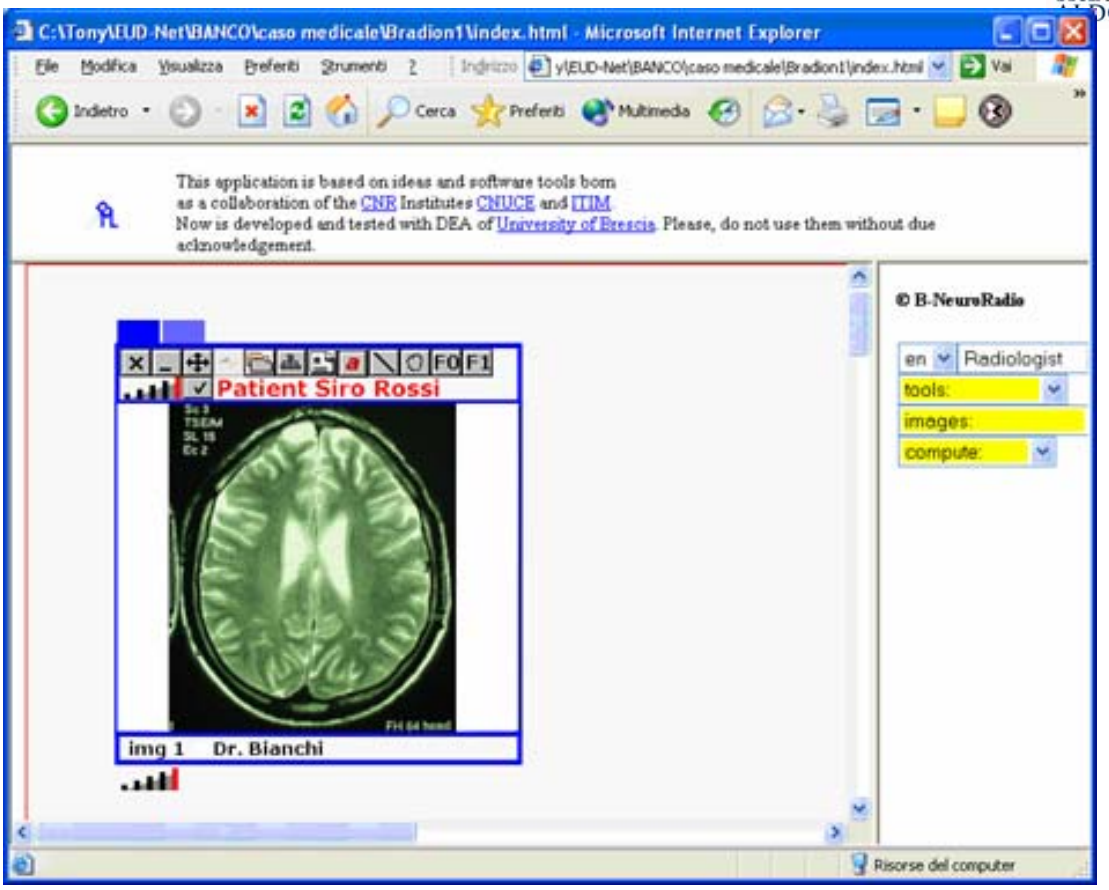
The medical scenario

- Techniques offer physicians *new possibilities*
- Working organization requires physicians to use new *technologies for collaboration*
- Different physicians collaborate working in different hospitals and having different expertise
- They overcome their communication hurdles by using electronic tools in a naïve way
- They required us to improve the quality of their communication, and therefore of their diagnostic capabilities

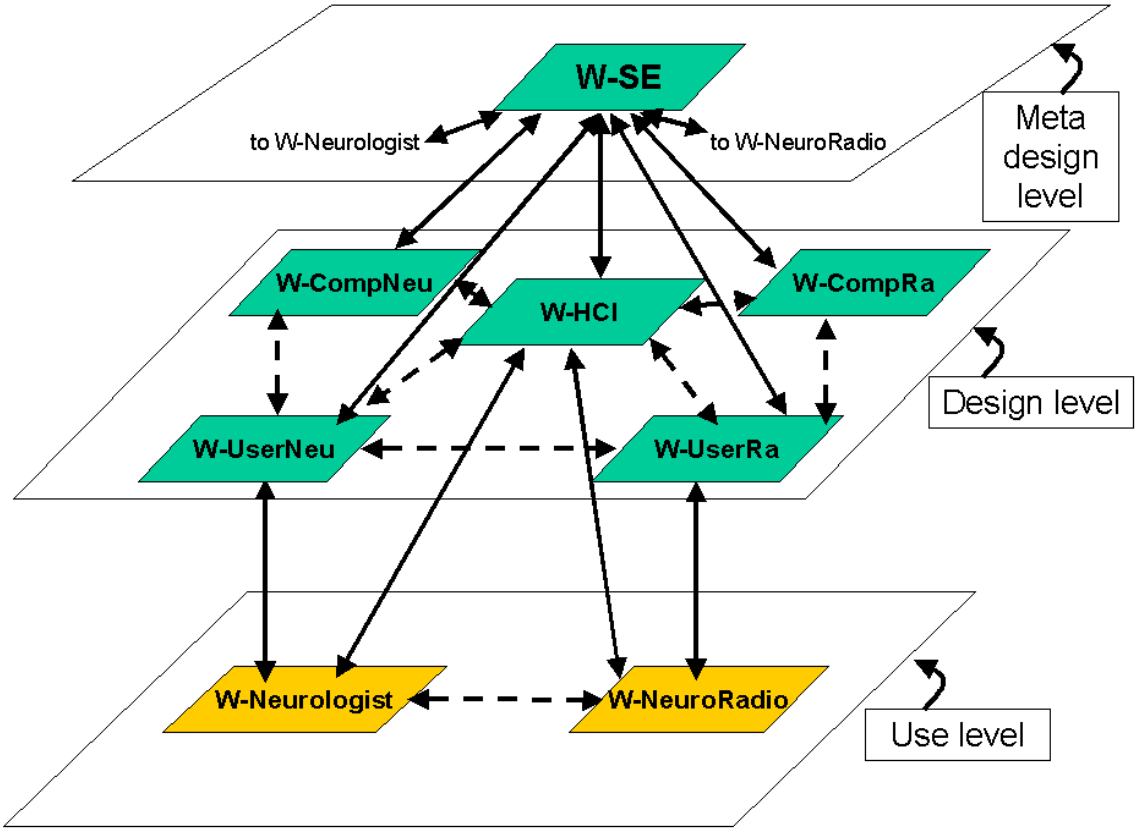
The scenario



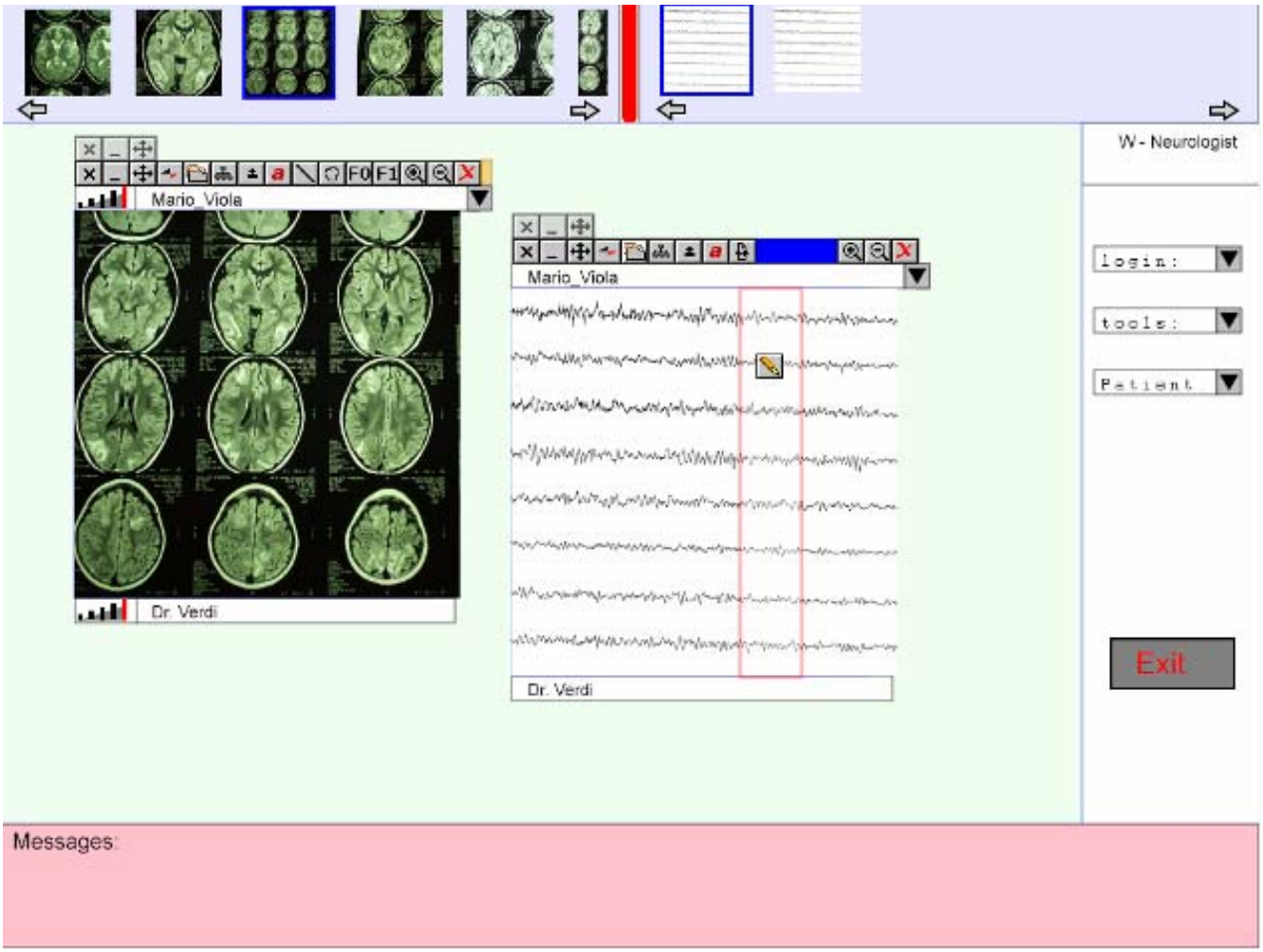
Primo prototipo SSW neurologo



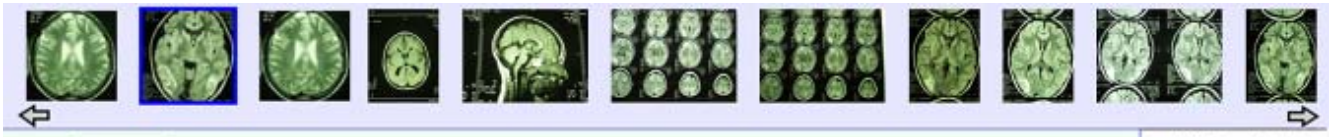
Rete di SSW nel caso medico

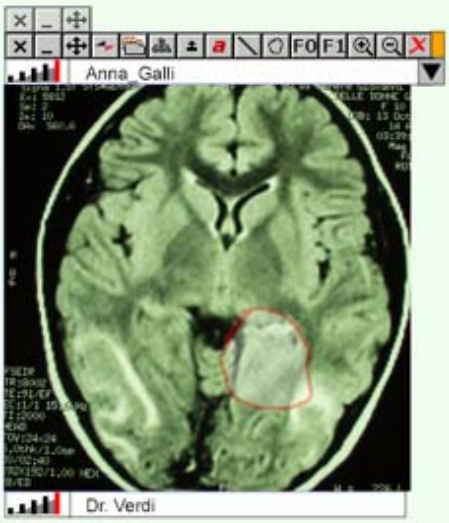


Secondo prototipo SSW neuro-radiologo

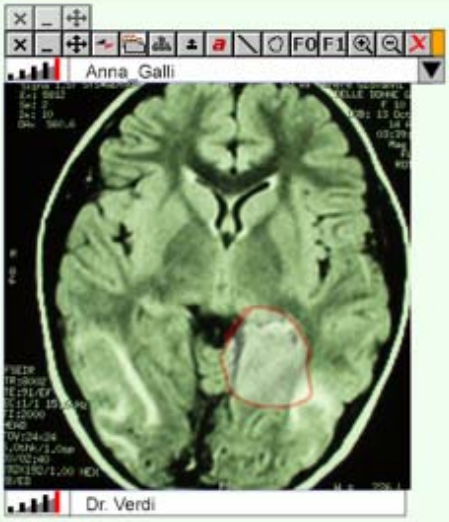

W - Neurologist
Mario_Viola
Dr. Verdi
Mario_Viola
Dr. Verdi
login:
tools:
Patient
Exit
Messages:

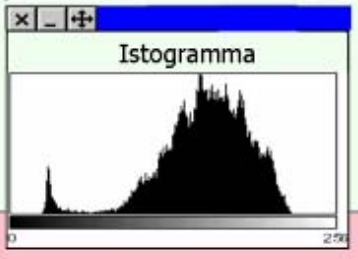
Iperglicemia alla nascita
Dolori addominali
Capo e occhi a sx
W - NeuroRadio
Anna_Galli
Dr. Verdi
Mario_Viola
nota 1/1
TITLE:
REPORT:
Add Nota
login:
tools:
Patient
Exit
Messages:



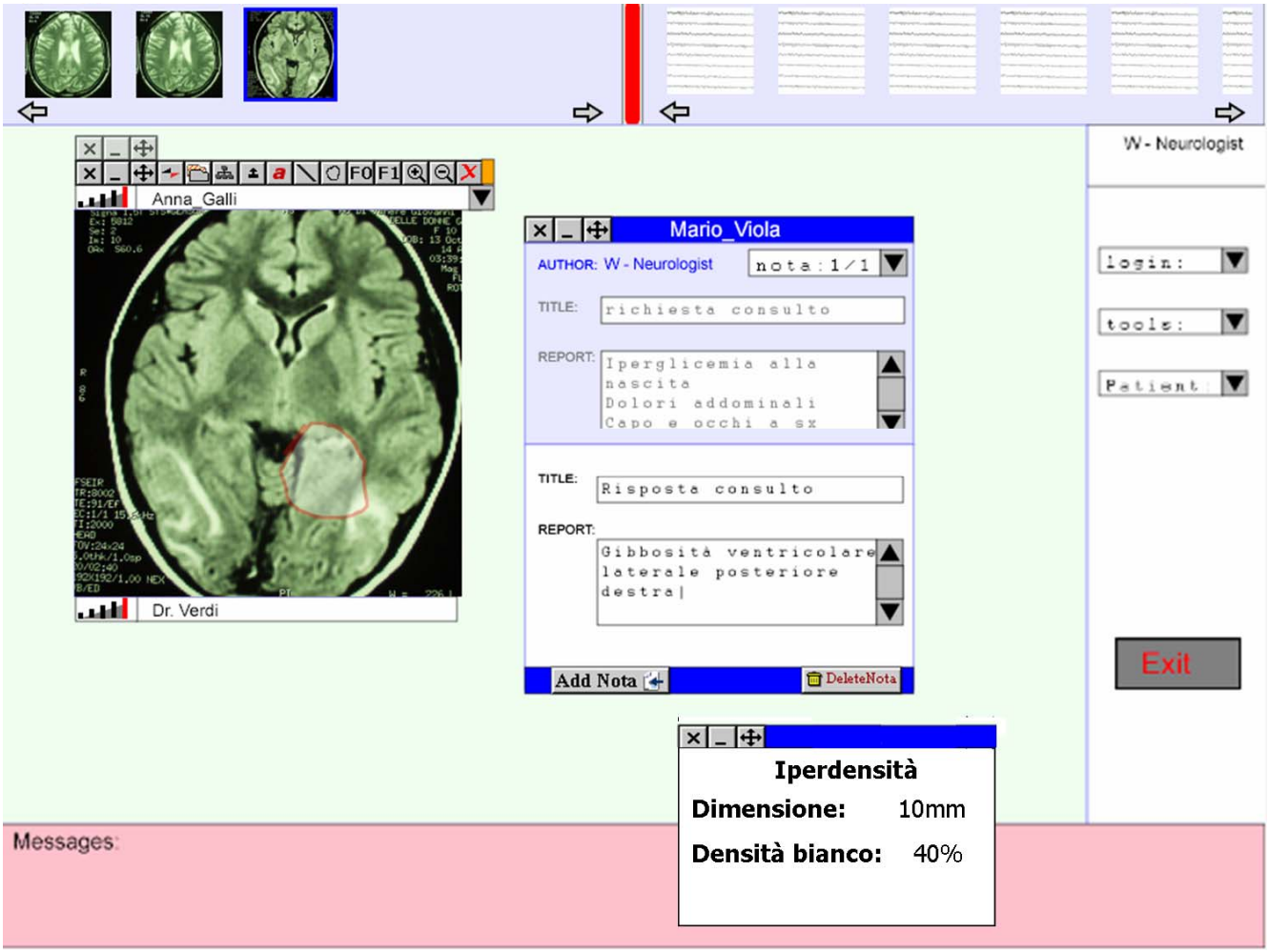


Messages:

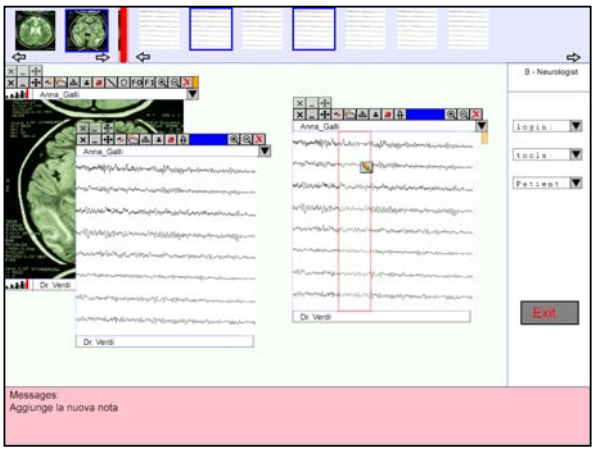




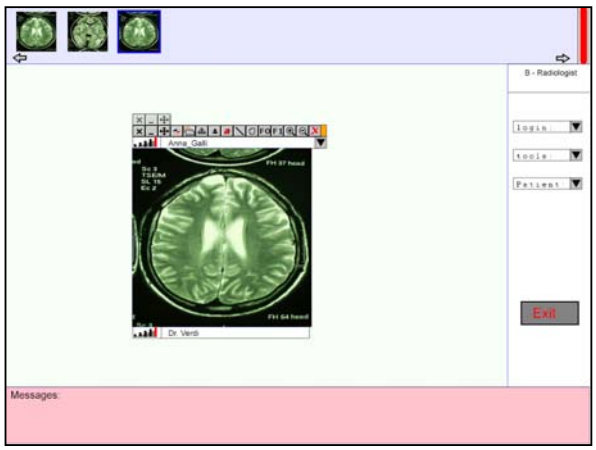
Messages:



The developed prototypes for neurologists and neuro-radiologists



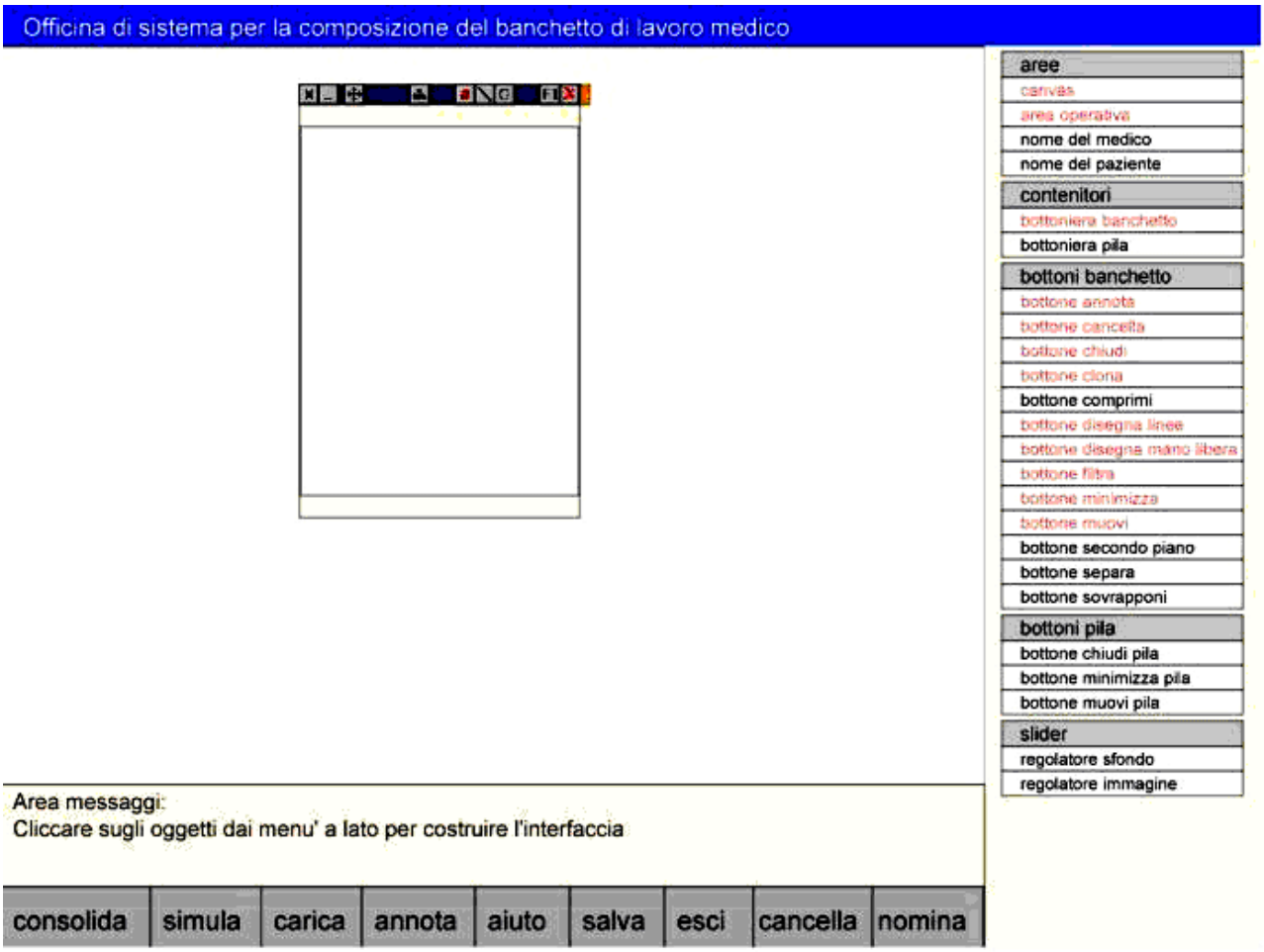
software environment for neurologists



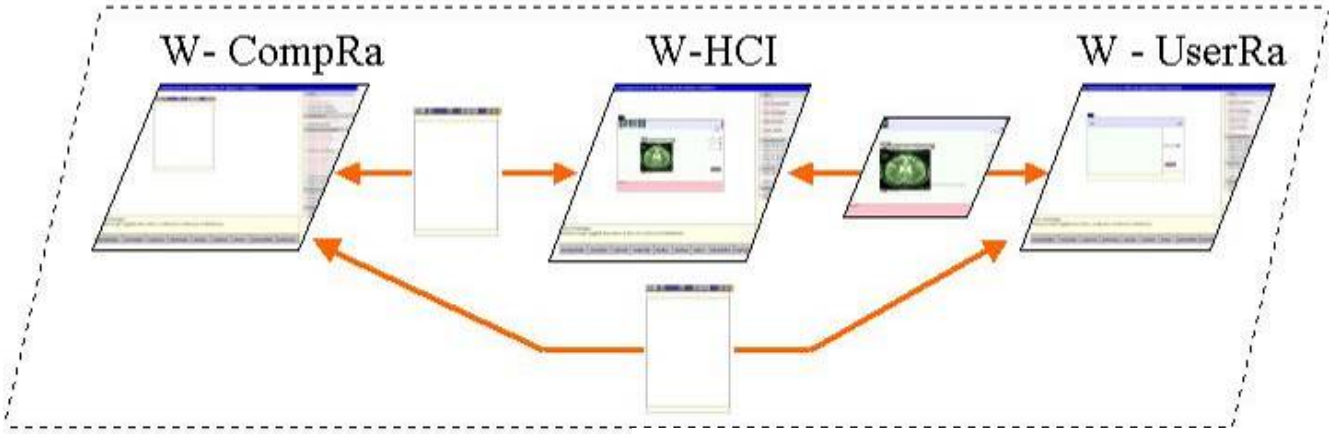
software environment for neuro-radiologists

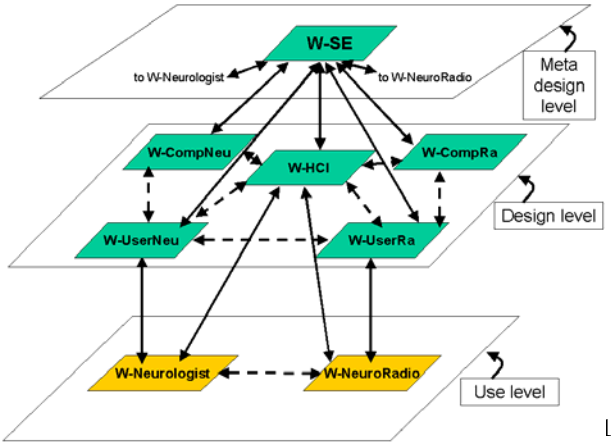
System adaptation to context

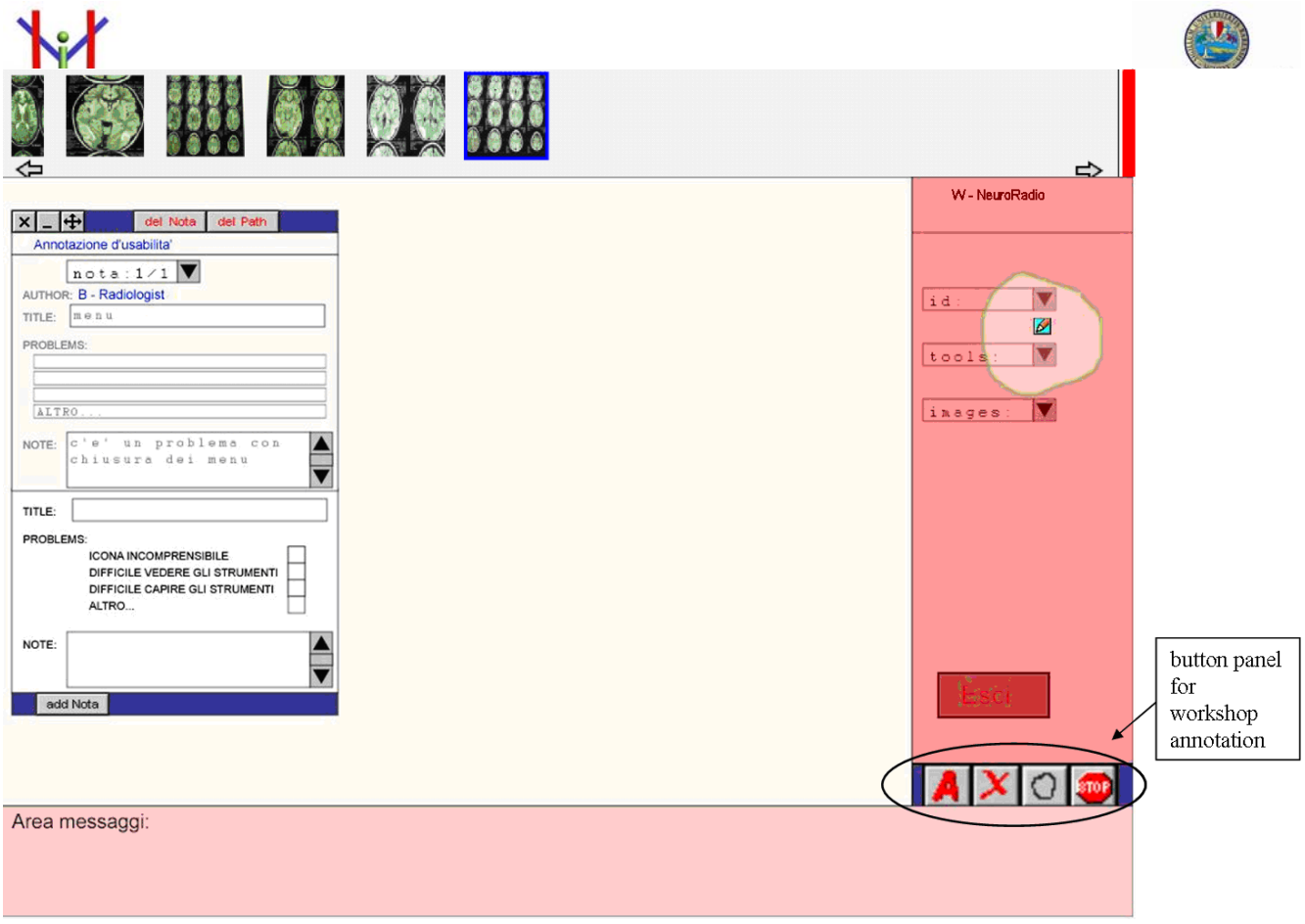
-The SSW associates the user defined widget with tools and materialization methods
suited to the physician's culture and skill



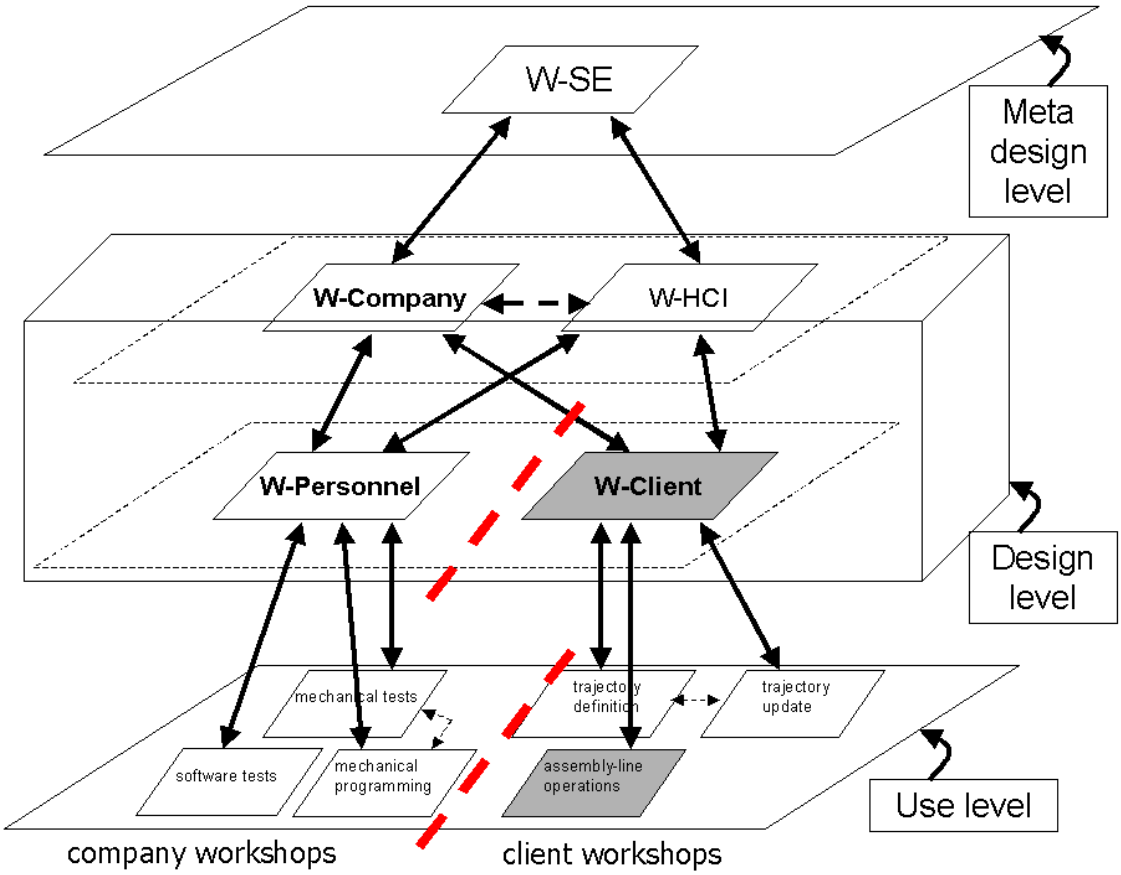
Comunicazione tra SSW

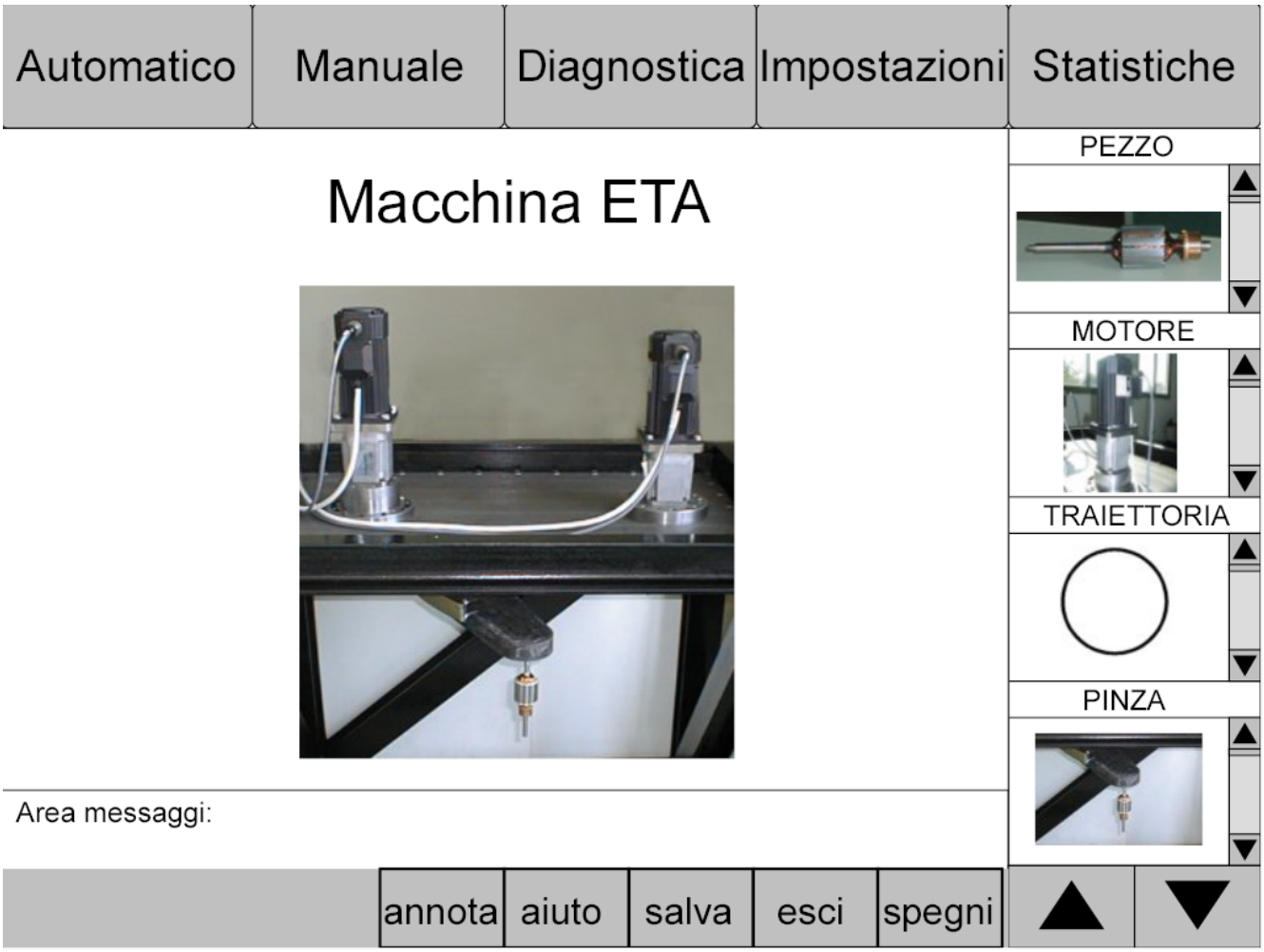




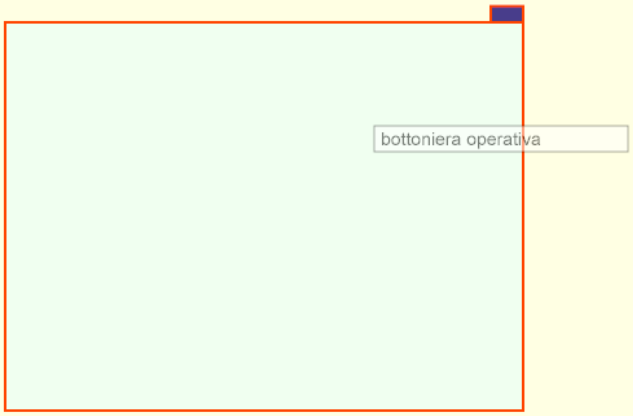


Rete di SSW nel caso meccanico





Officina di sistema per la composizione di interfacce



contenitori
- bottoniera di navigazione
- bottoniera di sistema
- bottoniera operativa
- area archivi

aree
- area operativa
- area messaggi
- canvas

bottoni di navigazione
- bottone automatico
- bottone manuale
- bottone diagnostica
- bottone impostazioni
- bottone statistiche

bottoni di sistema
- bottone annota
- bottone aiuto
- bottone salva
- bottone entra
- bottone spegni

archivi
- archivio pezzi
- archivio motori
- archivio traiettorie
- archivio pinze

Area messaggi:

Bottoniera operativa, va posizionata su di una canvas

salva officina | simula | carica | annota | aiuto | salva | esci

Officina di sistema per la composizione di interfacce

contenitori
- bottoniera di navigazione
- bottoniera di sistema
- bottoniera operativa
- area archivi

aree
- area operativa
- area messaggi
- canvas

bottoni di navigazione
- bottone automatico
- bottone manuale
- bottone diagnostica
- bottone impostazioni
- bottone statistiche

bottoni di sistema
- bottone annota
- bottone aiuto
- bottone salva
- bottone entra
- bottone spegni

archivi
- archivio pezzi
- archivio motori
- archivio traiettorie
- archivio pinze

Area messaggi:

Trascinare gli oggetti dai menu' a lato per costruire l'interfaccia

salva officina | simula | carica | annota | aiuto | salva | esci

Conclusions

- The SSW methodology is aimed at allowing
  - designers to create systems adequate for a user community, on the basis of users' knowledge, languages, and notations
  - Domain-experts users, not expert in Computer Science, to use and tailor their systems and to perform EUD activities
- Overcome phenomena that make HCI diffcult
- Gentle slope of complexity

Esempio di EUD all'Universita'

- A una riunione in cui si presentava la proposta di un portale per certi servizi, e' proprio venuta fuori l'esigenza di un utente di creare il suo report personalizzato.
  La risposta del progettista e' stata che ci deve essere il call center, al quale fare una richiesta telefonica di creare un nuovo report secondo le proprie esigenze.

- Un'attivita' di EUD dovrebbe mettere in grado l'utente di personalizzare il suo report.