Designing game logics for dynamic Active Surfaces

Gronvall Erik
Communication Science Dpt.
University of Siena
Via dei Termini 6,
53100 Siena IT
+39 0577 270565
gronvall@media.unisi.it

Pollini Alessandro
Communication Science Dpt.
University of Siena
Via dei Termini 6,
53100 Siena IT
+39 0577 270565
pollini@media.unisi.it

Rullo Alessia
Communication Science Dpt.
University of Siena
Via dei Termini 6,
53100 Siena IT
+39 0577 270565
rullo@media.unisi.it

Svensson David
Computer Science Dpt.
Lund University
Ole Römers väg 3
SE-223 63 Lund
+46 46 2224249
david@cs.lth.se

ABSTRACT

This paper describes the development of a modular system of interactive tiles to support therapists' in performing therapeutic activities together with impaired children in a swimming pool.

Active Surfaces support mobile interactions and dynamic configuration of assemblies of tiles. Each tile represents an interactive unit, able to communicate with other tiles and to exchange data. The tiles can be assembled on a physical and logical service level to support activities of different complexity. This creates technical challenges where assemblies are re-created over time between devices with limited input and output capabilities. Furthermore, Active Surfaces challenge the concept of understandability and how users can make sense of assembled systems with no or limited output capabilities.

Categories and Subject Descriptors

C.3 **[Special-purpose and application-based systems]** *Microprocessor/microcomputer applications, Process control systems, Real-time and embedded systems, Signal processing systems.*

General Terms

Performance, Design, Reliability, Experimentation, Human Factors.

Keywords

Ubiquitous computing, Palpable computing, end-user composition, distributed game logic, dynamic assembly.

1. INTRODUCTION

Mobile devices and small computer based systems can have a strong need for interaction, communication and information retrieval. These needs and how they are handled are not always clear in a standard ubiquitous setting. Traditionally, the notion of ambient computing has been consolidated focusing on the design of distributed, pervasive and reactive systems able to communicate with us and to continuously adapt to our current needs and expectations [1]. The exploration of smart objects and distributed systems highlight, from an interaction design perspective, subjects like devices, services, assemblies and distributed logic.

Ubiquitous and portable devices will be used in a wide range of situations, involving different physical and social environments and tasks. They will need to allow usage in many different ways and combination of modes. They should be configurable in the most appropriate way given the objectives, the users' competencies and the task at hand. Furthermore, unlike traditional ICT, users of mobile and pervasive devices will, typically, experience dynamic properties in use, for example higher levels of degradation and task switching.

Issues like inspection, of both breakdown situations and system configuration becomes important key-points in modular, mobile devices with limited interaction possibilities. These themes become crucial if applied to critical domains such as health care and rehabilitation. Active Surfaces tries to investigate these properties using an evolutionary approach on ambient computing with the notion of Palpability. Palpable computing aims at supporting user control by composing and de-composing assemblies of devices and services, while supporting meaningful interaction with users. These assemblies are configurable by the user depending on the context, the user's needs and expectations. Consequently, these systems should support continuous attribution and negotiation of meaning through interaction.

These challenges and others are addressed in the EU-funded Integrated Project PalCom: Palpable Computing – A new perspective on Ambient Computing (http://www.ist-palcom.org). This project uses the term 'palpable computing' to denote a new kind of ambient computing which is concerned with the above mentioned challenges in complex and dynamic ambient computing environments.

In this paper we will present a design concept supporting rehabilitation practice performed in a swimming pool embodying some of the qualities that creates the foundation of palpable computing. The practice built around rehabilitation activities performed in a swimming pool for disabled people serves as the main inspiration for this application.

At the public swimming pool in Siena (Italy), a volunteer association provides training and group activities for disabled people. The swimming pool by itself represents a powerful and privileged setting, because water supports the body and takes the weight off the joints making for example movements less painful. Moreover, water is a great 'equalizer' for disabled people: they find their movements easier and less different from those of non-disabled people while inside the water.

Today, different aspects of the rehabilitation practice are not well integrated with each other. Specific tasks and tools are designed for the motor physiotherapy; whereas other tasks, aids and tools are defined to support the acquisition of cognitive skills. The two activities are usually never integrated. You train one thing at the time, either A or B, rarely the two of them together.

Our aim is to join cognitive and physical rehabilitation objectives by providing the children with meaningful activity games supported by a distributed system of smart surfaces. The application system relies on the specification and reference implementation of a virtual machine for palpable devices, the PalVm, designed to match the evolving needs within the PalCom project [2]. The PalVm is a language-neutral virtual machine designed to support object oriented languages and small footprints. PalVm programs are deployed in binary components which are instantiated as run-time components, objects that encapsulate a set of classes and their required and provided interfaces. Applications running on top of the PalVm can take advantage of the palpable qualities offered by this platform. Active Surfaces is one example of such an application.

2. ACTIVE SURFACES

Active surfaces is thought of as mobile smart devices that serve for mobile gaming and rehabilitation activities. In this section the concept of Active surfaces will be explained.

2.1 Initial suggestions from fieldwork

An intense period of observations in the swimming pool permitted us to capture some peculiar features of the rehabilitation practice. We consider these suggestions as relevant for triggering our design process.

On-the-fly configuration. Dealing with continuously changing conditions and demands, the therapists need to find out creative solutions. They ask for adaptable tools to fit the needs of their patients even along the single treatment session. A core characteristic is that the tools have to be easily reconfigurable on the fly so they can adapt to these emerging situations.

Re-using games and keeping track of best practices. The therapists can rely on their experience and creativity in designing solutions. They would like to have the possibility to collect best practices and to re-use successful configurations of objects (devices) and games. Such a resource would contain also the know-how concerning alternative usage of the existing devices and services configurations.

Dealing with failures and degradation. The therapists usually deal with dynamic settings and changing conditions. This implies the ability to rearrange the available resources even in case of degraded performance of system. The tools they use have to guarantee robustness to the water setting (e.g. waterproof material and tight cases) and resilience in working functionalities. They would need also to recover from components failures by using substitutes devices to not interrupt the session.

Combining cognitive and physical rehabilitation. The lack of integration of physical and cognitive rehabilitation represents a constraint for current rehabilitation practice. The cognitive tasks are usually too static and children may lose attention. On the other hand, motor rehabilitation is very demanding at a physical level and is based on repetitive sequences of actions: patients often perceive them as tiring and not engaging.

2.2 Concept and components

The tiles act as building blocks that can be combined with a library of content (e.g. images, sounds and pictures). Furthermore they have reactive behaviours in relation to different input actions and orientation. Each tile provides outputs as visual or tactile feedback to support the accomplishment of the tasks given and to guide the patients in the interaction as described in section 4 below.

Today three kinds of Active Surfaces components exist: there is an Assembler Tile and a number of 'normal' tiles, and there is also a user interface, the Migrating User Interface (MUI)1 [3]. The PalVM is the platform on top of which these components run."

The ‘normal’ tiles are the ones used in the different activities and games together with the different users. Then there exists a privileged tile, the Assembler Tile. It is used by the therapists to program the other tiles. The Assembler Tile is equipped with a touch-sensor, realized by an infrared transmitter and receiver. This touch-sensor allows two interaction modes, a ‘one-touch’ or a ‘double-touch’ that evoke different functionalities in the Assembler Tile. These functions and their use will be described further down in this document. The MUI is used in a pre-activity phase to create general game logics and rules that later can be used in the pool. A therapist downloads the general rules created in the MUI into the Assembler Tile and brings it to the pool. Now the therapists can assemble different ‘normal’ tiles using the Assembler tile to support a wide range of activities. The tiles themselves once assembled constitute a network of physical (and software) objects that communicate and exchange data and are able to recognize their relative positions.

These features allow constructing meaningful configurations of different tiles. Each configuration is intended as an assembly of components. The therapists can configure these assemblies of components to define rehabilitation tasks. They can save successful configurations, keep memories of previous configurations and generate new assemblies to support patients’ specific needs. The rehabilitation activities enabled by the active surfaces allow a smooth integration of cognitive and physical tasks.

The Active Surfaces concept accounts for the need of configurability, constructability, modularity, physicality and creativity in rehabilitation practice. ‘One’ Active Surface consists of a tile, measuring 30*30 cm. Each Active Surface is thought of as a modular unit that can communicate with the others by its six sides. The tiles are able to recognize their relative positions in respect to other tiles. A number of tile

[1] The MUI End-User Composition Tool is used for programming the behaviour of the tiles. MUI, developed within the PalCom project (www.palcom.dk) at the University of Lund, enables a therapist to browse existing tiles and their configurations. She can discover the tiles currently in use, and design a new exercise in the MUI browser (e.g. running on a PDA or a laptop).

components can be assembled to constitute a network of physical (and software) objects that communicate and exchange data. Many qualities of palpable devices are embodied in the Active Surfaces concept. Today a prototype is being used for evaluation purposes based upon a Basic Stamp 2 micro controller and IR communication. These tiles (as seen in figure 1) offer limited functionality, but sufficient for initial trials and proof-of-concept. The next generation of tiles embedding the full vision and the palpable framework is currently under development together with the University of Aarhus (Denmark) and Lund University (Sweden).

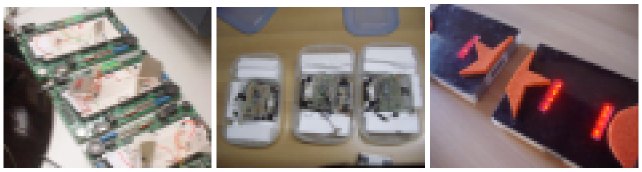

Figure 1. Initial working prototype under construction and final result

The main idea is to rethink the environment of the pool, making it a place for rehabilitation and play activities. Today the swimming pool is designed for swimming: the water serves as the mean of interaction. People usually don't have any (strong) relation with the pool by itself: the edges and the bottom are not conceived for any purpose of interaction. The design process aims at re-considering the surfaces of the pool and to change the activities that usually take place there [4],[5].

As interaction designers our focus is on enabling environments and tools supporting engaging rehabilitation activities. The Active Surfaces is the concept that embodies these issues. The surface of the pool becomes active re-designing the bottom, the edges and even the water surface. In this vision the floating tiles constitute one of the main supports for the interaction and the therapeutic activity.

3. PALPABLE QUALITIES

Active surfaces represents an exemplar application of the palpable computing framework being developed in the PalCom project. Limited resources and basic input- output profiles characterize these devices.

Palpability emerges as a property-in-use of the tiles' assembly. Dynamics of physical- logical construction/deconstruction, mobile interaction and services communication sum up the palpable qualities of the application.

Being conceived as an assembly, active surfaces could provide a valuable example of physical construction/deconstruction of components. Thus the physical construction of assemblies [2] provides end-users with control of the system behaviour and adaptation to the context. As Active Surfaces don't rely on data and information visualization; the tiles assume physical construction and physical interface as major strengths for the users.

The Active Surfaces constitute assemblies on different levels: on the logical level the therapist can define what the rules are and what the purpose is. On the functional level the user can mark out the relations and the sequences and on the physical levels the user can define which patterns and connections can take place, to reach the final solution.

By focusing on the virtues of therapist (end-user) composition, the Active Surfaces is also complemented by the MUI browser mechanism for programming the rules and the behaviours to be instantiated in the tiles. The therapists then create patterns by physically building tiles' sequences [6]. The tiles address also scalability, offering the opportunity to produce scalable solutions still relying on low level resources management. Palpable systems can be described, as many ambient computing system, as a heterogeneous mix of distributed, embedded devices with different capabilities. The Active Surfaces must provide scalability and persistence across different devices so that errors in one part the system do not spread to other parts of the system. These features are supported by an appropriate degree of decoupling between different parts of the implementation.

By adopting labels and grouping of the devices, Active Surfaces have chance to survive through system degradation: when interruptions occur the users can substitute components while continuing the activities.

The system can still guarantee understandability also concerning the balance between system automation and therapist' (i.e. user) control. The tiles have to preserve the understandability and support the users to maintain control over the technology.

Regarding usability of resources and resource access, the Active Surfaces create flexible ad-hoc networks connecting the single devices. Networked tiles preserve their own identity and dynamically seek for available tiles in the vicinity. The tiles continuously inspect what communication processes are taking place at the moment looking for specific connection on all its sides.

The proof of the Active surfaces concept may also enable the designers to explore the relation between change/stability in use. In fact the assembly' behaviours are instantiated in physical configurations that can be saved, reused (also in part) and instantiated in different physical patterns. The therapist can show the right pattern (sequence) to the system and record (save) the configuration by using an assembler tile. The dynamics between configurations' change and stability may address the future practice of rehabilitation and the way in which the Active Surfaces could support it.

4. SCENARIO

Along the different phases of the work analysis we used scenarios to evaluate, together with our stakeholders, how the defined concepts could suit their needs and to envision possible usage of the final tools. Scenarios themselves were used as design objects and they evolved along the design process being created, refined and also sometimes dismissed [7].

As far as we have developed the early prototype, we also designed games of different kinds and of different degrees of complexity, which can easily show the potentials of the application. The basic activity Active Surfaces can be used in is a position-based game. The position games regard all the sequence-composition activities where users have to place letters, pictures, textures or colours in the right order following the activity task. For instance the position game we give as an example in the scenario below is based on the composition of the word "C I A O". This is a linear, sequential word construction. In this kind of games the tiles have a unique ID and they know exactly what is their identity and position in the sequence.

The more complex kind of games can be described assuming the Scrabble task as main example. Providing multiple simultaneous combinations, the scrabble is such a meaningful setting in which the same letters are present in a number of occurrences. For instance, considering many words, it is possible that the letter E occurs several times and each E should know what is its correct position and orientation in the crossword assembly. Thus all the letters E need a label that group them, through which they can better identify their role in the game and have to receive the right behaviour for the accomplishment of the task. The Scrabble game logic will be described in par. 4.2.

The games we have been designing with Active Surfaces follow a logic based on condition satisfaction rules. Tiles' states are described through the use of a "happiness" state. These terms are used with specific meanings in the scenario and in the code development. We consider different states of happiness (conditions' satisfaction) for the position and orientation of the tiles in the assembly.

- *SideHappiness* means that a tile realizes that it is correctly connected on a particular side. On the side(s) that are Happy the tile provide the users with HappySide feedback. If all its sides are correctly aligned, *LocalHappiness* is instead achieved.
- *LocalHappiness* means that the tile is properly connected to the others and it has on each side the tiles it was looking for. It is in the right position and it is correctly orientated in the assembly.
- *UnHappy* – A tile is unhappy when it lacks *LocalHappiness* (i.e. while no happiness state or only *SideHappy* is reached, a tile is still *UnHappy*). While a tile is *UnHappy* it broadcasts its *UnHappy* state to the other tiles.
- *AllHappiness* means that all the tiles satisfy the *LocalHappiness* state. A tile that is not *UnHappy*, does not communicate anything to the other tiles. This gives, the lack of *UnHappy* messages (given a certain timeout) within the system together with *LocalHappiness* realizes the *AllHappiness* state. The game is solved.

4.1 Active Surfaces in use

The therapist or trainer configures the activity outside the pool; at her home or in a remote office. She can also configure the activity in the vicinity of the pool, but there is no specific need for that from a system perspective.

The following PalVm services enable this early phase of configuration:

- *PalCom device discovery* The device discovery enables a device running the PreWM to autonomously discover new (palpable ready) devices in the vicinity.
- *Service discovery* As the Device discovery was used to automate the discovery of devices, the Service discovery is used to query and identify running services on remote devices.

Furthermore is each tile aware of its own unique id, a dynamic label and can propagate this data through its 6 sides. The tiles can also listen on incoming traffic on their 6 sides once the PalVm has communicated via the device discovery and service discovery that there is another tile present on a particular side.

4.1.1 Configuring the activity

The therapist attaches the assembler tile (AT) and the MUI Browser with each other. The PalCom device discovery now allows the MUI and the assembler tile to dynamically connect and exchange data. The PalVm then uses the service discovery to locate the correct or required services. Once these services have been identified, behaviours and game rules can pass from the MUI to the AT. The therapist configures the activity by setting those parameters in the tiles that constitute a game or therapeutic activity, such as the kind of activity (e.g. Position game or Scrabbles game) and the output mode (e.g. Blinking light).

In a first version of Active Surfaces the AT will have a cable connection (i.e. RJ-45 Ethernet) between the system hosting the MUI and the AT. This connection will later on be wireless.

4.1.2 Configuring the game tiles

The therapist can now bring the AT to the pool and align the tiles she would like to include in the game, showing physically the tiles the 'winning' position and pattern. This activity will initiate device and service discovery activities inside the tiles. A tile now knows which tiles are connected to its sides, and their orientation as illustrated in Figure 2.

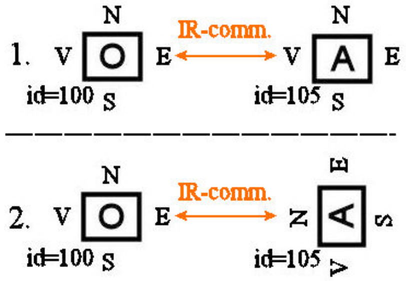

Figure 2. Row 1. Tile id=100 knows that on side 'E' it has 'V' side of the tile id=105 and vice versa.

Row 2. Tile id=100 knows that on side 'E' it has 'N' side of the tile id=105 and vice versa.

The AT is now being connected to the sequence of tiles. The therapist can rely on a simple physical interface on the AT: by using 'one touch' input she settles the tiles in the sequence. This command sends a broadcast message to the connected tiles to set their neighbour's ID and positions. This means that each tile looks which other tiles are currently connected to its sides and saves this information together with the orientation of these neighbour tiles.

The broadcast message can appear as follows:

AT to all Tiles: Set Neighbours(){ Set Identity & Orientation }

Now all the tiles have memorized their own and their neighbour's ID, position and orientation as shown in figure 2.

The feedback the tiles have to produce, when reaching the different Happy-states are now being downloaded from the AT to all the tiles.

The tiles now notify the therapist when the configuration is successfully achieved by providing the settled output. Since they are now in their winning-position (i.e. they are all in a *LocalHappiness* which gives an overall *AllHappiness* state, this since no *UnHappy* messages are being broadcasted).

At this point the AT asks for the current assembly of tiles, to store the current assembly for later use as a trace of best practice to serve as an inspiration for other therapists or patient activities.

The therapist is now ready to start the activity with the patient and throws the tiles into the pool. The child now starts to play with the Active Surfaces. The initial positions can be viewed in figure 3.

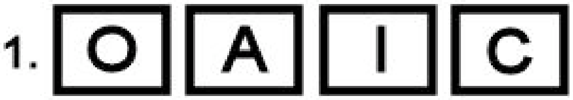

Figure 3. Initial configuration, the tiles are thrown into the pool. There is NoHappiness.

A *Discovery report (on a PalVm level)* takes place as soon as the game starts. Each tile starts to report which tile is close to its sides constantly matching these data with the stored *Neighbours* data.

In the early tentative trails the child tries different wrong alternatives by moving around the tiles. This provokes new *Discovery* actions (on a PalVm level). When a tile is recognized on a side, a getNeighbour action is performed (on a service level). This gives the tile information of the Identity and Orientation of the tiles surrounding it. The message the different tiles broadcast is: *AllUnHappy*. This since they are all in a non *LocallyHappy* state.

Step by step the child finds out local solutions for the tile game but still not the global solution that solves the complete sequence.

The child puts two tiles aligned following the right configuration, but still not with the complete solution presented. This gives a local feedback that the two tiles are correctly placed while the final feedback is still not given as shown in figure 4.

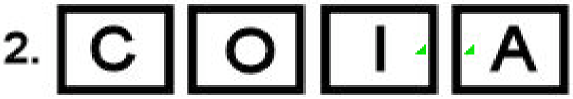

Figure 4: HappySide for 'I' and 'A' between the 'I' and 'A' sides (Green 'triangle' feedback).

Get Neighbors() (C, O, I, A) gives *HappySide = true* for *I.East* and *A.West*. This gives that I and A enters a *HappySide* mode on these sides.

The connected sides' lights on I and A are now being turned on and (green light) feedback is given to demonstrate that the *HappySides* condition on these sides have been reached. For both the tiles the *LocalHappiness* state is not yet true because it doesn't realize all the required conditions for their happiness state.

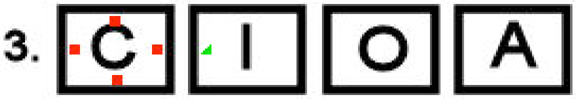

Figure 5: 'C' is LocallyHappy (Red 'square' feedback) and 'I' has one HappySide (Green 'triangle' feedback).

Figure 5 shows another try to reach the game solution. Given that the winning solution is the word sequence "CIAO", C and I are in their right position and orientation. C now reaches a complete *HappySide* state and thus is *LocallyHappy*. This *LocallyHappy* state provides the user with a (red) light output depending on the initial configuration.

Get Neighbors() (C, I, O, A) gives *HappySide = true* for *C.East* and *I.West*. These condition makes *C=LocallyHappy* and *I=SideHappy*.

The global state of the active surfaces is maintained by a broadcast of the message *UnHappy()* from those tiles that are not LocallyHappy. That puts all the tiles in a *AllUnhappy* state.

Going toward the final (and global) solution of the game, the child starts to align all the tiles in the right position by following the linear orientation of the word sequence.

Moving all the tiles correctly gives the final output and the game is solved.

getNeighbors(C, I, A, O) gives *HappySide = true* for *C.East-I.West* and *I.East – A.West* and *A.East – O.West*. This results in that all the tiles (C, I, A and O) reaches the *LocallyHappy* state.

When all tiles are in the LocallyHappy state (i.e. no *UnHappy* message is broadcasted) the tiles reach the *AllHappiness* state. The game is solved as presented in figure 6.

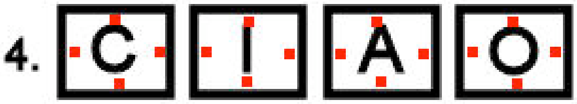

Figure 6: All tiles are LocallyHappy, this gives a complete happiness within the system (AllHappy).

4.2 Scrabble game logic

Scrabble[2] game is a popular word board game, in which two to four players score points by forming words from individual lettered tiles on a 15x15 game board [8]. The words are formed across and down in a crossword fashion, and must appear in a standard dictionary. Active Surfaces can be used in high complexity scrabble-like game in which each word crosses the others and has letters in common. In the example given we describe the composition of "H E L L O" and "L O O K" and how they can create the scrabble assemblies that appears in figure 7.

[2] "Scrabble" is a registered trademark of Hasbro, Inc

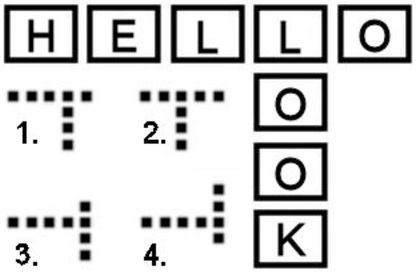

Figure 7. Solution 1 and 2 can be reached using only the tiles illustrated in the big picture, if an extra L is present, solution 3 and 4 can be reached. However, leaving a spare O tile.

Being present three L and three O, it is necessary to characterize the tiles' identity and to distinguish their position and orientation in the sequence.

Starting a new scrabble game activity, the therapist can create labels with a *SetEqual* command by using the Assembler Tile. This creates a unique *tileLabel* for the group of tiles currently attached to the assembler tile; in the example shown in figure 8, the L group and the O group have been created before the game. The therapist performs this action by the *Double touch* command on the AT interface setting the label equal in the group of tiles. Now the grouped tiles still have different IDs but share the same instructions for the connections (position and orientation of the neighbouring tiles). For instance the three L can be used in the different positions still maintaining adequate coherence in the game.

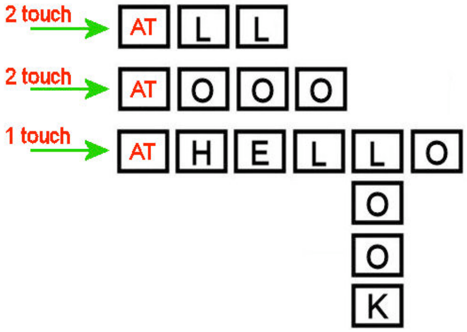

Figure 8. The Assembler Tile is used to group the tiles and create labels.

Grouping the tiles is the mean through which users can manage tiles' identities and eventually recover in case of tiles' breakdown by substituting them with new ones. When created in a group, the tiles under the same label might substitute each other and the therapist would be allowed to recover from system degradation at-hand.

In this way Active Surfaces shows a high combinatory power to support the therapist in game composition creation along her practice.

4.3 Re-assembling the game and storing

During the activity the therapist can re-attach the AT and change the game. She is now configuring a new assembly to be created along the activity with the patients. When this is done without the Browser, the game owns the same input/output profile as in the existing activity. It is nevertheless allowed to configure the tiles populating the game by differentiating her identity or positioning, just by placing them differently in relation to the others.

The assembler tile also provides the therapist with the opportunity to store and re-use previous games. When connected to the tiles along the setting up of the activity (e.g. *SetNeighbours and SetEqual*) the AT records and stores the configurations that are being used and the shapes created with the tiles. When the AT is re-connected to the MUI Browser, the stored data are uploaded to the MUI and stay available for the different therapists and trainers. The database has been created both as a memory of the past activity and as an inspirational repository of use (e.g. Best Practices collection).

5. DISCUSSION

The concept of Active Surfaces elaborates distributed game logics for the rehabilitation practice in the swimming pool.

The scenarios we developed are based on the idea of end-user composition, mobile interaction and user control and it seems very promising for our stakeholders at the hospital and at the swimming pool. In particular, users appreciate the idea of being supported by ready-at-hand and easy to program technology. The Active Surfaces provides them with the possibility to improve the day by day rehabilitation practice.

The concept elaborates on a new challenging view of construction complemented with deconstruction of physical assembly. The therapist is asked to manipulate and physically configure the tiles while the dynamic and self-configuring discovering of components occurs. This guarantees minimum skills in technology and programming for the users.

Active Surfaces provides the therapist with the possibility to adapt the technology pursuing extreme change and flexibility while keeping system stability. In that way we have situations where total control is desirable, but has to be complemented with sense making and meaning attribution of events.

Palpable computing can be seen as extending ambient computing with additional characteristics for user control. Palpable computing systems offer not only invisibility (the capacity of unobtrusively performing computing tasks in the background environment) but also visibility, that is, the capacity of making visible to users what they are doing and what they may do. Moreover, systems should offer both construction (the ability to support end-user composition of devices or services to form new devices and/or services) and also deconstruction, that is, the ability to disassemble a device or service into its constituent parts to enable understanding and manipulating of each part individually [2].

Palpable computing may constitute the framework for the definition of design guidelines for mobile interactive systems. Challenges in the shape of dichotomies as the ones described above, can serve as means through which the designer can interpret and re-define pervasive applications.

6. ACKNOWLEDGMENTS

The work regarding the game-logic and the communication between the tiles has been developed together with University of Aarhus (Denmark) and Lund University (Sweden). For their work in relation to this paper we would like to especially thank Prof. Boris Magnusson and Dr. Görel Hedin.

Our thanks also go to Laura Cardosi, Maria Grazia Burroni and the therapists at the Functional Rehabilitation Unit of the "Le Scotte" hospital, Siena, for their open-minded collaboration and continuous support during fieldwork and participatory design. The research was part of the EC IST PalCom project.

7. REFERENCES

[1] Weiser, M. The Computer for the Twenty-First Century. *Scientific American*, pp. 94-10, September 1991.

[2] Schultz, U.P., Corry, E., Lund, K.V. Virtual Machines for Ambient Computing: A Palpable Computing Perspective *OT4AmI ECOOP Workshop Position Paper* 2005.

[3] Svensson,D. Magnusson,B. and Hedin,G. Composing ad-hoc applications on ad-hoc networks using MUI *Proceedings of Net.ObjectDays, 6th Annual International Conference on Object-Oriented and Internet-based Technologies, Concepts, and Applications for a Networked World*, Erfurt, Germany, September 2005.

[4] Greenbaum, J. Kyng, M. *Design at Work*. Hillsdale, NJ: Erlbaum, 1991.

[5] Marti, P., Rizzo, A. Levels of design: from usability to experience. *HCI International 2003, 10th International Conference on Human - Computer Interaction*, Crete, Greece, Lawrence Erlbaum Associates. 2003.

[6] Gronvall, E., Marti, P., Pollini, A. Rullo, A, Active Surfaces: a novel approach for end-user composition *NordiCHI, Fourth Nordic Conference on Human- Computer Interaction* 2006.

[7] Rizzo, A., Bacigalupo, M., Scenarios: Heuristics for action *Prooceedings of XII European Conference on Cognitive Ergonomics, D.J. Reed, G. Baxter, M. Blythe (eds.)* 2004.

[8] Wikipedia http://en.wikipedia.org/wiki/Scrabble