Supporting Distributed Private and Public User Interfaces in Urban Environments

Simo Hosio[1], Marko Jurmu[1], Hannu Kukka[1], Jukka Riekki[2], Timo Ojala[1]
[1] MediaTeam Oulu, Department of Electrical and Information Engineering
[2] Intelligent Systems Group, Department of Electrical and Information Engineering
University of Oulu, Finland
firstname.lastname@ee.oulu.fi

ABSTRACT

Proliferation of large public displays in urban cityscape gives rise to applications distributed between public displays and mobile devices. However, real deployment of distributed applications on top of this new infrastructure is challenging as no commonly accepted architectural solutions exist to rely on. In this paper, we present a platform supporting distributed application user interfaces on interactive large public screens and personal mobile devices. We demonstrate the functionality and potential of our approach by presenting a deployment of the platform with multiple distributed applications in authentic setting in a city center. We found this platform feasible to deploy interactive, appealing services on top of, and a non-cost information pick-up service as the most appealing to users.

Categories and Subject Descriptors

H.5.2 [**Information Interfaces and Presentation**]: User Interfaces – *Prototyping; user-centered design; user interface management systems (UIMS).*

General Terms

Design, Experimentation, Human Factors, Verification.

Keywords

User interfaces, urban computing, experimentation, system design

1. INTRODUCTION

As the proliferation of large public interactive displays in urban cityscape is advancing, various service scenarios utilizing this additional screen real estate with mobile devices arise. However, real deployment of distributed applications on top of this new infrastructure is challenging as no commonly accepted architectural solutions exist to rely on. User interface (UI) design and HCI issues are often regarded as key research areas in pervasive computing. As wireless network technologies are maturing to provide the required mobile high speed access to online services surrounding us, as well as privacy and security for dealing with private data in mobile devices [1], the deployment of functional and safe distributed user interfaces (DUI) [2] into cityscape - out from the traditional research laboratory environments and into the hands of everyday users is becoming reality. Day by day, our digital lives move increasingly into personal mobile devices, thus making them natural choice of central control point for surrounding services as well. We live among many platforms and devices, but UI systems architectures surrounding us are still bound to the old-fashioned "one screen, one input" paradigm [3]. This needs to change in order to make progress in the field of modern mobile and distributed UIs.

HotMobile 2010, February 22–23, 2010, Annapolis, Maryland,US.

Previous research on issues related to distribution of UIs between mobile devices and resources offered by the environment can be categorized based on the layer governing the distribution:

Migratory user interfaces, discussed thoroughly in [4], are interfaces capable of traversing among different devices and platforms according to mostly system inferred rules and continue their task there, pertaining the internal state and functionality.

UI rendering and language abstractions revolve around device-independent GUI specification languages or models, studied extensively in [5], and presenting these GUI compositions with platform-specific rendering engines by system-inferred rules to provide high-level device-independency. A good practical example is the combination of HTTP, CSS and web browsers.

Application composition [6] can be regarded as one of the DUI categories as well. Application composition architectures commonly make independent decisions on physical and logical composition of the application and its logical parts including the presentation layer during runtime rather than design-time or compile-time. Applications in composition architectures commonly have policies and rules to determine the best possible outcome for the resulting composition of the application. However, application composition can also be user-centric, as recently studied in [7].

We argue that personal mobile devices will increasingly begin to dominate access to resources embedded to our everyday environments. This trend is also seen in the development of various input techniques to contemporary mobile terminals [8]. These factors combined suggest that personal mobile terminals are ideal control points also for shared interactive public displays, effectively alleviating previously documented issues such as social awkwardness [9] or physical placement problems [10].

We present a solution that bundles together mobile devices, application servers and public displays to form a DUI and a platform for designing and building appealing services. We support applications with both *private UIs for control on mobile side* and *public UIs on large public screens.* The spectrum of application possibilities supported by this platform is wide, even without complex system rules required by many DUI solutions. We illustrate this later with multiple example applications. We discuss the DUI solution of the architecture and its feasibility in deploying urban applications to genuine end users. Our platform supports touch screens, or any other interaction technology used on public side, as the developer is free to choose any public web based UI and mobile interfaces residing on top of our lightweight J2ME layer to integrate the service with the rest of the architecture. This platform does not anyhow govern the input or output mappings of UIs, but rather manages the physical division of the UI components between the public and private resources.

The most crucial design issues of distributed applications supported on top of this platform are separation of the application components and I/O logic in design time by the developer, and the composition of the application host devices, performed run-time by the user. These raise research challenges such as allocating and scheduling the public display resources, run-time user centric composition of host devices, and run-time personalization and localization of applications.

The contribution of this work is as follows:

1. A DUI platform capable of facilitating urban mobile services with four different user interface distribution models, including distributed control points of an application to both private and public UIs.
2. Ability to support on-the-fly instantiation of both private and social multi-user applications with distributed UIs deployed on mobile phones and public displays.
3. Bottom-up user-centric composition of applications and especially the user interfaces of applications as perceived by the end users.
4. Deployment and empirical evaluation of the proposed platform and distributed applications in authentic setting in a city center.

An example service utilizing the offered DUI functionality is ***PlaceMessaging***. It allows leaving messages directly from mobile phones on a "virtual cork board" situated in the public displays, such as seen in Figure 1. Messages contain text, emoticons, and photos. Later on, users can view, arrange, and play with them on the public UI. Thus, PlaceMessaging utilizes both private control on mobile side and shared, public touch-screen interface on shared public displays.

Figure 1. A large public display at downtown Oulu.

The rest of this paper is structured as follows. Chapter 2 presents the design and implementation of the DUI functionality with fundamental design and technical requirements for distributed applications. Chapter 3 introduces example mobile applications utilizing the different UI models supported. Chapter 4 discusses early findings and statistics from our field trial. Finally, we discuss the strengths and weaknesses of the proposed platform in comparison to other existing solutions, summarize our contributions, and lay path for future research.

2. DESIGN AND IMPLEMENTATION

Our platform is designed and implemented as a part of a large urban computing test bed deployed in downtown Oulu, Finland. This ongoing experimentation focuses, among others, on interactive usage of mobile devices with large public screens deployed in real life setting. This platform has been verified by a field trial during summer 2009. We pursue the flexibility and ease of implementation of distributed user interfaces in the environment. Our test bed supports various urban cityscape elements from large public displays with touch screen capabilities to smart phones, various different types of context sensors, service discovery mechanisms, and event based control messaging between devices. However, sensor infrastructure, service discovery and traffic routing are out of scope of this paper.

2.1 Conceptual Model

The DUI platform presented here is a component in a larger software and hardware infrastructure called *The Open Urban Service Network* (OUSN). The OUSN consists of distributed resources - most importantly large public displays, different sensors, heterogeneous network access, distributed communications, an API to support application/service development and service discovery mechanisms. It is designed as extendable, and implemented with open source software component where possible. The OUSN is presented in depth in [11]. The model of supporting applications with DUIs in the OUSN realm is presented in Figure 2.

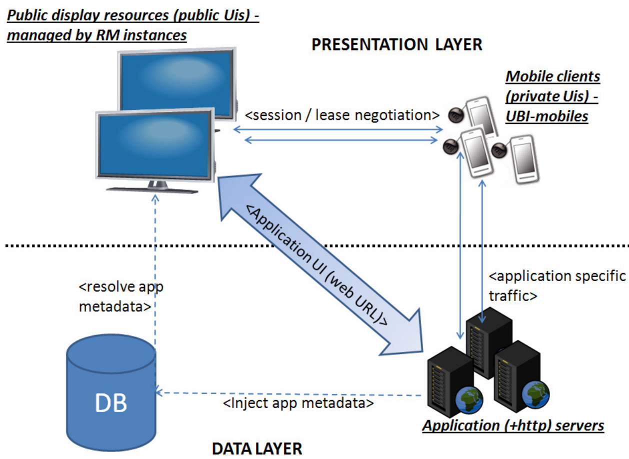

Figure 2. The distributed UI support in the OUSN. Mobile clients handle leasing of the displays and application servers provide the application's public UI to the display resources.

Our platform supports four different types of distributed user interfaces:

1. **Control on mobile UI, public UI as a one-way display.**
2. **No control on mobile UI, public UI as control point, utilizing touch screen or other interaction techniques.**
3. **Control on both UIs, mobile and public screen.**
4. **No specific control points, application is plain distributed display on mobile and public UI.**

This platform does not aim to migrate the states of UIs or applications, or represent and render the same user interface on different platforms and devices, but rather supports the inherent creation of applications with dedicated DUI parts, private mobile UI and public UI, which can be interactive if desired. The application composition is decided in the design phase instead of relying on arbitrary runtime composition rules as in the composition architectures.

2.2 Components

ResourceManager (RM) component acts as the central coordinator of a public display and its currently displayed UI [12, 13]. It exchanges control events with the mobile clients, negotiating the permissions for application sessions and triggering the actual public user interface to be displayed on large display when needed. During the launch of a public UI, the RM provides the UI with extra parameters, namely the current application user's or users' friendly names and the friendly name of the place where the display is physically located. This way, the UI has possibility to personalize the visual appearance to some extent.

Application sessions are represented by **leases** [12], which can be either social or private, as defined by the application metadata. The type determines whether the session allows the participation of multiple users in the same application or by just one single user at a time. Thus, we provide support for "arena" type of applications that allow anyone to collaborate and participate as well as for private, single user applications.

UBI-mobile is the mobile access point and service discovery interface of the architecture. The UBI-mobile displays the available services in a certain context and handles the negotiation of leases with the RM responsible for a public display. UBI-mobile's service discovery screen and basic look and feel are illustrated in Figure 3. Service discovery mechanisms are outside the scope of this paper, but currently we are utilizing, among others, an RFID-based physical user interface to access services [14]. The UBI-mobile launches services via the MIDP 2.0 Push Registry (http://developers.sun.com/mobility/midp/articles/pushreg/).

UBI-MIDlet is the light software layer underlying the mobile part of services launched by UBI-mobile. UBI-MIDlet is a J2ME layer that provides native service support by inheriting it from the standard J2ME MIDlet application framework. It implements session control, transparent integration with the server components, authentication and other functionality needed in the OUSN infrastructure. The UBI-MIDlet's responsibilities include taking over the control of a lease from the UBI-mobile during the startup of a service, thus each application always controls one lease (social or private). Upon session termination, the application built on top of the UBI-MIDlet is responsible for ending the lease by calling a shutdown method of the parent UBI-MIDlet.

Applications are distributed entities realizing one of the four user interface types introduced earlier. Each application is identified by a unique ID located in its metadata, ranging from 0x0000 to 0xFFFF. ID is needed e.g. when negotiating leases and retrieving the rest of the application metadata from a database. Metadata consists of various attributes, such as one defining service's multi-usability. Namely, applications can be **social or private**, similarly to leases. This defines the distribution model of the public user interface towards multiple clients. Social applications are by nature capable of serving multiple clients simultaneously. Private applications serve only one client at a time. The public UI can be anything rendered by modern web browsers; AJAX-page, Flash-application, Applet, etc., thus offering a rich development platform. We have used Firefox 3.0 web browser in a kiosk mode for presenting the public UI. RM has also built-in support for launching native program binaries, but in this work we have utilized web based UIs only. On mobile side we use the J2ME user interface components. As the natural UI element support of the J2ME is rather limited, we utilize J2ME Polish toolkit (http://www.enough.de/) to offer developers novel UI support. Applications can include arbitrary application server functionality – the division of computation is up to the developer.

Figure 3. From top left: Idle UBI-mobile, discovered service set available for launching after user has been granted a lease to a public display, private UI of a poker application, and at bottom: screenshot of the public UI with two participants.

2.3 Resource Access and Scheduling

Figure 4 illustrates the control sequence to enable run-time composition of applications with DUI and scheduling application sessions among multiple client devices. We employ a bottom-up user-centric application and UI composition mechanism, where the device composition is first assembled via leasing the displays through RFID-based physical interaction, which is seen as a good way to access services in smart environments [14].Then the distributed application to be launched on top of this device composition is discovered and chosen. If there are no sessions currently, a lease is granted. If the application running is social, the user joins the current application session. Otherwise, if a private application is already occupying the display resource, additional clients enter a FIFO *lease queue*. The first client in the FIFO queue is notified by audio effect and vibration of the mobile phone once the public resource is free to be used by the client. This approach comes handy when the applications are run in places where people tend to spend more leisure time, such as a café or pub, and might thus be comfortable with waiting for their turn to use the shared public display.

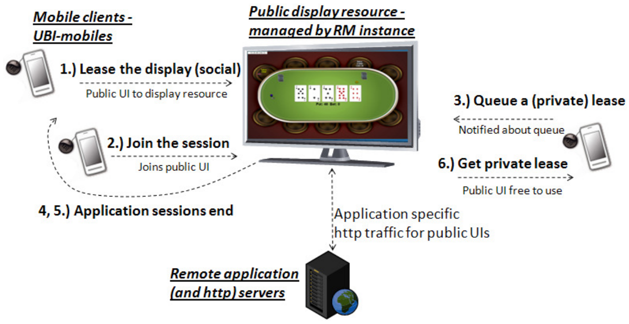

Figure 4. Mobile clients queue for a UI lease: After the ongoing session (social) with two clients ends, the display resource is leased to the device in the queue (on the right).

2.4 Application Development

When designing a distributed application for this platform, certain guidelines have to be followed. The design begins with the division of the application UI to private and public parts. As we are concentrating on large display surfaces, the public UIs are not suitable for presenting highly personal data. This naturally has to reflect in the application design itself [9]. Technically, the public UI can be designed for any modern web technology, or even a binary application. The design of the private part is currently constrained by the J2ME platform. This restriction, however, will be addressed in our future work. If the application being designed aims for participation and input from the audience and bystanders, the designer needs to acknowledge large public displays experiencing inherent lack of attention from the general public [15]. Indeed, one of the biggest challenges in utilizing public displays for distributed applications is developing appealing and engaging content.

3. EXAMPLE APPLICATIONS

We present six applications built using our platform, to illustrate the alternative ways of combining the public and private (mobile) UI parts into different distributed applications. Figure 5 illustrates which of the four different distributed UI models each of the applications employs.

	NO MOBILE CONTROL	HAS MOBILE CONTROL
NO PUBLIC CONTROL		Social Surroundings UbiRockMachine UBI-Poker
PUBLIC CONTROL	BlueInfo UBI-gallery	PlaceMessaging

Figure 5. Categorization of the example applications according on their DUI models.

Social Surroundings [16] is a social application which pulls user created social media content from popular third party services to enhance communal behavior and conversation in a public space – a concept proven attractive also in [17]. Other application examples using this UI model are ***UbiRockMachine*** [18] for community voting of audio playlists and ***UBI-poker***, a social Texas hold 'em poker game played with public displays and mobile phones (see Figure 3). These applications employ the private (mobile) UI for personal and private control of the application and a large public screen for presenting the public UI to users and spectators alike. ***BlueInfo*** [19] allows users to download various data directly from a public touch screen interface to their mobile phones via non-cost Bluetooth technology. No control is provided in the mobile UI, but it contains usage instructions for quick reference. The public UI facilitates selecting the data, which is pushed to the mobile UI for simple information pickup. Thus, BlueInfo follows the second DUI model, just like the ***UBI-gallery***. It allows the users to tag and delete photos they have previously uploaded to one of our gallery applications from the native image galleries in their mobile phones. A key strength of our platform is the support of services with control points on both the private mobile UIs and the public UI. **PlaceMessaging** discussed already in introduction serves as an example of this DUI model.

4. EVALUATION

The platform and the applications have been empirically validated in authentic setting in a city center. We deployed 11 so-called *UBI-hotspots,* in five indoor and six outdoor locations around downtown Oulu. Since the double-sided outdoor hotspots contained two large public displays back-to-back, we had 17 public UI's in total. We loaned out preinstalled mobile devices to test users so that they could use the displays and services. General public could also install the mobile client to their own phone. The first trial lasted for three months in June-August 2009. Here, we analyze data gathered during a period of 46 days from July 17 to August 31. While services were available to citizens during the whole three-month field trial, July 17 was the first day when all 11 hotspots were functional.

We gathered both quantitative and qualitative data using different methods including semi-structured interviews, observation, questionnaire and automatic logging of every interaction event in each hotspot.

4.1 Qualitative Findings

Over the course of the data-gathering period, 81 semi-structured interviews were conducted. Many of our interviewed users found the access to the services via the UBI-mobile's service discovery rather clumsy as they were accustomed to starting services directly from their phones. Thus, they found UBI-mobile just rather useless even though we presumed it to provide added value in form of context-specific applications in a given hotspot. However, the UBI-mobile is currently a necessary component for our DUI infrastructure to function. Downloading free content such as timetables, weather and general news with BlueInfo was particularly successful application. Test users found it easy to learn and suggested many new types of content to be available via BlueInfo, such as movie timetables and restaurant menus or specials.

The idea of distributing UI between the mobile device and the public displays seems easy to adopt, but the challenge lies in getting users to try out mobile services instead of the plain touchscreen services. Social Surroundings [16], for example, is a prime example of a service that is quoted as very entertaining and fun after use, but it is challenging to guide the user through the necessary steps to start using and experimenting with the service. Furthermore, the physical location of the display plays a big role in an application's chances to become popular. Information pick-up works better in busy passageways whereas social applications benefit from deployment in cafes and casual places where people have more idle time. In our earlier tests, we found Texas hold 'em as a very appealing social application on top of our platform in such places.

4.2 Quantitative Analysis

During the data-gathering period we logged 726 mobile application sessions. When using a social application, each of the clients joining the application session was regarded as a new session. Interestingly, the BlueInfo information pick-up service turned out to be the most used application with 385 sessions. This indicates the need of retrieving relevant and especially up-to-date mobile information in urban settings. The next most often used applications were PlaceMessaging and UBI-gallery with 166 and 144 sessions. The average duration of an application session was rather short 1:33 minutes, which we presume to be due to the placement of the hotspots. Most of the hotspots are located on busy passageways, thus they are more suitable for short sessions and quick interactions. In any case, this data is probably sufficient to verify the functionality of the proposed DUI mechanism and the possibilities it offers for deploying distributed applications to the general public.

5. DISCUSSION

Distributed user interface research is at an interesting turning point as new computationally powerful display resources are rapidly emerging in urban areas. Thus, the traditional laboratory experiments could be deployed in areas occupied by ordinary citizens and city dwellers. However, the majority of the theoretical work on distributed user interfaces lacks the potential and maturity to be really deployed on the public displays in urban environments and to be used by devices carried by the general public,. We have taken a more concrete approach deploying our system in an authentic setting in a city center for 24/7 use by the general public, and automatically gathering valuable research data in the background. Thus, we have a good opportunity to examine the usage and impact of a real life DUI deployment.

Regarding user interfaces with mobile devices and public displays, research has been largely focused on how and which kind of content to represent, and how to interact with it in public, semi-public or private spaces [9, 20, 21], interaction techniques with the displays themselves [10], and privacy and security issues with public UIs [22, 9]. Furthermore, distributed interfaces comprising of large displays or projection surfaces and mobile phones have been employed to stimulate audience participation [23] or to create artistic expressions [24, 25]. Mostly these prototypes, while contributing to user interface research, approach the DUI paradigm via a short-term deployment in a controlled semi-public environment. Our long-term installation in a city center aims at the general public adopting it so that it someday would be an integral part of their daily urban lives, hopefully.

5.1 Limitations

Currently, our DUI platform suffers from some practical shortcomings. The current mobile support is based on a J2ME layer, which is needed for various UI-independent features of the OUSN framework. However, we are currently redesigning the framework to address this.

The public UI sets limitations for application developers as well. We cannot allow developers to freely deploy native applications in the control PCs of the public displays, as they would pose a serious security and possibly a performance issue to our system. Using web based public UIs, we gain better control of the application UI and its possible misbehavior by monitoring the visual state of the public UI. This way, we're also able to banish the application UI to return the display to its default state. Further, the web based public UI facilitates simple integration of any 3rd party application in the public Internet into our hotspots.

To initiate the negotiation of a lease to a public display, users have to physically touch the embedded RFID reader even when the display is in use, which can be perceived socially awkward for some users.

5.2 Conclusions

We presented a DUI platform, which has been deployed in authentic setting at downtown Oulu, Finland. We presented six example applications which have been implemented on top of the platform and discussed preliminary findings and statistics from the first field trial.

Our key contribution consist of supporting mobile applications with four different DUI models, supporting on-the-fly usage of private and social multi-user applications with DUIs, and a bottom-up user-centric composition of applications and especially the user interfaces of applications.

The platform allows building of engaging and appealing services ranging anywhere from information retrieval and social networking to place-specific messaging or multimedia applications. So far the most interesting mobile application according to the users has been the BlueInfo information pick-up service.

5.3 Future Directions

To tackle the platform restrictions of the mobile clients, we have redesigned the functionality of the *UBI-MIDlet* to be offered via a proxy server and a common protocol such as SOAP. Thus, a platform-independent signaling channel will be offered for applications wanting to utilize our DUI platform, enabling application development for any given modern mobile device.

Currently, the only option to lease a display is through RFID interaction. A Bluetooth-based solution is under development, where a registered device within the proximity of the display is able to lease a display. Communication will take place via the aforementioned signaling channel, but Bluetooth discovery will serve as the proximity identifier.

We will expand the hotspot deployment into casual places where people tend to spend more leisure time, such as cafés, lounges, and pubs. In these places, the separation between private and public UIs offers various research possibilities on especially social and communal applications, such as Social Surroundings [16] and UbiRockMachine [18]. An interesting aspect regarding these public installations is not only the added value perceived by the end-users, but as well the added value contributed to the business owner of the space. For a sustainable and long term evaluation, our deployment must be accepted and administrated by the host, instead of our research staff.

We are also planning to completely open the OUSN infrastructure and invite both academia and industry to prototype and exhibit their applications.

REFERENCES

[1] H Sharp, R., Madhavapeddy, A., Want, R., and Pering, T. 2008. Enhancing Web Browsing Security on Public Terminals Using Mobile Composition. In Proceeding of the 6th international Conference on Mobile Systems, Applications, and Services, MobiSys '08. ACM, NY, 94-105.

[2] Larsson, A., Berglund, E. 2004. Programming Ubiquitous Software Applications: Requirements for Distributed User Interface. In Proc. of the 16th International Conference on Software Engineering & Knowledge Engineering (SEKE '04), Banff, Anbert, Canada, 246-251.

[3] Olsen, D. R. 2007. Evaluating User Interface Systems Research. In Proc. of the 20th Annual ACM Symposium on User interface Software and Technology. UIST '07. ACM, NY, 251-258.

[4] Berti, S., Paterno, F., and Santoro, C. 2005. A Taxonomy for Migratory User Interfaces. In Proc. of the 12th International Workshop on Interactive Systems, Design, Specification, and Verification, Newcastle upon Tyne, UK, 149-160

[5] Nylander, S. 2003. Approaches to Achieving Device Independent Services - an overview. Technical Report T2003-16, SICS 2003.

[6] Mazzola, P, J., Pham, H., Saif, U., Chau, G., Terman, C., and Ward, S. 2008. Structured Decomposition of Adaptive Applications. Pervasive and Mobile Computing 4(6), 791-806.

[7] Newman, M. W., Elliott, A., and Smith, T. F. 2008. Providing an Integrated User Experience of Networked Media, Devices, and Services through End-User Composition. In Proc. of the sixth International Conference on Pervasive Computing, Sydney, Australia, 213-227.

[8] Ballagas, R., Borchers, J., Rohs, M., and Sheridan, J. G. 2006. The Smart Phone: A Ubiquitous Input Device. IEEE Pervasive Computing 5(1), 70-77.

[9] Huang, E., Koster, A., Borchers, J. 2008. Overcoming assumptions and uncovering practices: When Does the Public Really Look at Public Displays? In Proc. of the 6th International Conference on Pervasive Computing, Sydney, Australia, 228–243.

[10] Cheverst, K., Fitton, D., Taylor, N., and Kray, C. 2008. Experiences of Supporting Local and Remote Interaction with Touchscreen Based Situated Displays. In Proc. of Workshop on Designing and Evaluating Mobile Phone-Based Interaction with Public Displays at CHI 2008.

[11] Ojala, T and Kukka, H. 2009. A Digital City Needs Open Pervasive Computing Infrastructure. In Proc. of Digital Cities 6: Concepts, Methods and Systems of Urban Informatics, State College, PA.

[12] Jurmu, M., Perttunen, M., and Riekki J. 2007. Lease-Based Resource Management in Smart Spaces. In Proc. of the 5th IEEE international Conference on Pervasive Computing and Communications Workshops, NY, 622- 625.

[13] Jurmu, M., Boring, S., and Riekki, J. 2008. ScreenSpot: Multidimensional Resource Discovery for Distributed Applications in Smart Spaces. In Proc. of the Fifth Annual International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services (MobiQuitous '08), 1-2.

[14] Riekki, J., Salminen, T., and Alakärppä I. 2006 Requesting Pervasive Services by Touching RFID Tags. IEEE Pervasive Computing 5(1), 40-46.

[15] Müller, J., Wilmsmann, D., Exeler, J., Buzeck, M., Schmidt, A., Jay, T. and Krüger, A. 2009. Display Blindness: The Effect of Expectations on Attention towards Digital Signage. In Proc. of the 7th International Conference on Pervasive Computing 2009, Nara, Japan, 1-8.

[16] Hosio, S., Kukka, H., and Riekki, J. 2010. Reclaiming Our Shared Public Spaces Bridging the Virtual and Physical through Social Networking and Media. IEEE MultiMedia 17(2).

[17] Farnham, S. D., McCarthy, J. F., Patel, Y., Ahuja, S., Norman, D., Hazlewood, W. R., and Lind, J. 2009. Measuring the Impact of Third Place Attachment on the Adoption of a Place-based Community Technology. In Proc. of the 27th International Conference on Human Factors in Computing Systems, Boston, MA, USA (CHI '09) ACM, NY, 2153-2156.

[18] Kukka, H., Patino, R., and Ojala, T. 2009. UbiRockMachine: A Multimodal Music Voting Service for Shared Urban Spaces. In Proc. of the 8th International ACM Conference on Mobile and Ubiquitous Multimedia, Cambridge, UK, Article no.9.

[19] Kukka, H., Kruger, F., and Ojala, T. 2009. BlueInfo: Open Architecture for Deploying Web Services in WPAN Hotspots. In Proc. of the 2009 IEEE International Conference on Web Services, IEEE Computer Society, Washington, DC, 984-991.

[20] Vogel, D. and Balakrishnan, R. 2004. Interactive Public Ambient Displays: Transitioning from Implicit to Explicit, Public to Personal, Interaction with Multiple Users. In Proc. of the 17th Annual ACM Symposium on User interface Software and Technology. UIST '04. ACM, NY, 137-146.

[21] Kray C., Galani A., and Rohs M. 2008. Facilitating Opportunistic Interaction with Ambient Displays. In Proc. of Workshop on Designing and Evaluating Mobile Phone-Based Interaction with Public Displays at CHI 2008.

[22] Garnham, D., Buchanan, G., and Jones, M. 2008 Incidental Mobile Information and Large Public Displays. In Proc. of Workshop on Designing and Evaluating Mobile Phone-Based Interaction with Public Displays at CHI 2008.

[23] Scheible, J. and Ojala, T. 2005. MobiLenin - Combining a Multi-Track Music Video, Personal Mobile Phones and a Public Display into Multi-User Interactive Entertainment. In Proc. of ACM Multimedia 2005, Singapore, 199-208.

[24] Scheible, J., Ojala, T., and Coulton, P. 2008 MobiToss: A novel gesture based interface for creating and sharing mobile multimedia art on large public displays. In Proc. of ACM Multimedia 2008, Vancouver, Canada, 957-960.

[25] Scheible, J. and Ojala, T. 2009. MobiSpray: Mobile phone as Virtual Spray Can for Painting BIG Anytime Anywhere on Anything. Leonardo 42(4), 332-34.