Journal of Universal Computer Science, vol. 19, no. 17 (2013), 2526-2542
submitted: 15/2/13, accepted: 22/4/13, appeared: 1/11/13 © J.UCS

Design Considerations for Application Selection and Control in Multi-user Public Displays

Constantin Taivan, Rui José, Bruno Silva
(Centro Algoritmi, Campus Azurém, University of Minho, 4800-058 Guimarães, Portugal
{constantin, rui}@dsi.uminho.pt, brcpsilva@gmail.com)

Ivan Elhart
(University of Lugano, Via G. Buffi 13, 6904, Lugano, Switzerland
ivan.elhart@usi.ch)

Jorge Cardoso
(Escola das Artes / CITAR – Universidade Católica Portuguesa, Porto, Portugal
jorgecardoso@ieee.org)

Abstract: Urban spaces are increasingly embedded with various types of public digital displays. Many of these displays can be subject to multi-user interactions and support a broad range of applications. A fundamental implication emerging from the interactive nature of those applications is that users should have access to appropriate selection and control techniques that would allow them to drive the way applications are shown and used in the respective environment. Such techniques should enable each user to reason and express intentions about the system behavior, while also dealing with concurrent requests from multiple users in a way that is fair and clear. In this study, we aim to inform the definition of novel techniques for application selection and control in pervasive display environments that can address the above challenges. Drawing inspiration from traditional GUI interaction concepts we developed and deployed a public display system that supports multiple applications and is able to receive explicit content presentation requests from multiple viewers. Based on the experiment observations and interviews with the participants, we reached a set of design considerations for future pervasive displays environments that are open to third party applications providers and allow the audience to influence content presentation.

Keywords: Multi-application displays, multi-user interaction, mixed initiative approach, public displays.
Categories: D.2.1, D.2.10, H.1.2, I.2.0, K.4.2

1 Introduction

Urban spaces are increasingly embedded with ubiquitous computing technologies and in particular with various types of public digital displays. This is leading to the emergence of pervasive display systems that can be described as perch/chain sized ecosystems for many-to-many interaction, composed of displays of various sizes (from handheld devices, to medium/large wall mounted displays), and where "many people can interact with the same public screens simultaneously" [Terrenghi et al., 09]. Displays in pervasive display systems are inherently multi-purpose [Kostakos et al., 13] and increasingly becoming the subject of multi-user interactions. Our vision is to allow the emergence of open displays networks in which any entity in the network

can add to the value proposition of the entire system and where large-scale networks of pervasive public displays and associated sensors are open to applications and content from many sources [Davies et al.,12]. In these scenarios, content should no longer be distributed in the form of tightly pre-defined schedules, as it is current practice with existing digital signage systems. Instead, specialized web applications could provide that content, while being able to tailor their behavior with the information space associated with each public display and any local interactions from nearby people.

A fundamental implication emerging from the interactive nature of those applications is that users should have access to appropriate selection and control techniques that would allow them to drive the way applications are shown and used in the respective environment. Such techniques should enable each user to reason and express intentions about the system behavior, while also dealing with concurrent requests from multiple users in a way that is fair and clear. In this study, we aim to inform the definition of novel techniques for application selection and control in pervasive display environments that can address the above challenges. These should enable multiple users to concurrently drive the selection of the applications being shown and control their behavior.

To pursue this goal, we have devised a research methodology involving two phases. In the first phase, we extended our previous informative study on application selection and control techniques from existing GUI systems [Taivan et al., 12]. This provided us with a set of well-established GUI techniques for dealing with similar issues when considered for public displays. In the second phase, we analyzed how these techniques could be adapted to the specific context of public displays interaction and we created an implementation to support their evaluation in real setting.

Based on our field experiment observations and interviews with the participants, we reached a set of design considerations for future pervasive displays environments that are open to third party applications providers and allow the audience to influence content presentation. While all the participants reported the easiness and fairness of the application selection approach, they also noticed the need for more responsiveness and rich notifications for the various human-computer interaction stages, both for public and private/mobile devices.

The paper is organized as follows. After discussing related work [Section 2], we examine application selection and control mechanisms in traditional GUI systems and analyze to what extent these mechanisms could be appropriated in public display systems [Section 3]. Afterwards, we briefly describe the design and implementation of a multi-application display system; introduce the study on assessing the multi-user experience of influencing the presentation order of applications and present the results of the study [Section 4]. Finally, we conclude the paper in [Section 5].

2 Related Work

Public displays have largely been investigated from the perspective of single user interaction paradigm. In this model, one user at the time can appropriate a large display by accessing rich interactive applications and services [Ojala et al., 12][Hosio et al., 10][Davies et al., 09]. Displays as multi-user and shared resources have also been explored mainly from three perspectives: time based queuing, explicit space

partitioning, and implicit space partitioning. The e-Campus system [Davies et al., 09] employed a mechanism in which users' content requests were handled using preemptive priority queues. Based on a FIFO policy the display system presented the content for a pre-determined amount of time per each item. A similar approach is considered by [Pablo et al., 06], which focused at solving the conflicts in ambient intelligent environments.

Dynamo [Izadi et al., 03] consisted of a public display system for multi-user interaction where users could share content and control the graphical interface using wireless mice and keyboards. They could also explicitly "carve" rectangular regions of the screen and appropriate that region for personal or mediated use. Morales-Aranda et al. [Morales-Aranda and Mayora-Ibarra, 07] describe a display prototype as an implicit space partitioning system that can dynamically adapt content layouts allowing two users to visualize their personal information. Vogel et al. [Vogel and Balakrishnan, 04] also use an implicit space partitioning approach where the space allocated to each user grows and shrinks according to the number of interacting users. The Tacita system [Kubitza et al., 12] also uses a similar approach in their *Weather* application where the preferences of several users are combined and the application presents an integrated weather forecast. This functionality reflects the same assumptions that we are considering in our research, i.e., the display is a shared resource and is not solely appropriated by one user at a time.

A common way to interact with public displays in a single user as well as multi-user environment is through personal mobile devices. In the e-Campus system [Davies et al., 09] researchers used SMS and Bluetooth device names to interact and personalize the screens. The Ubi-Hotspot system [Ojala et al., 12] allows the users to interact with the displays using RFID enabled mobile devices. Similarly, Boring et al. [Boring et al., 13] use mobile devices to enhance limited input capabilities of public displays at different viewing distances. The camera of the mobile device is used as an universally deployed sensor to estimate the position of the user (and their mobile device) with respect to one or more public displays in the environment. Using this technique, the users can project on the large display: a) the entire device's screen and b) a touch input point from their mobile device screen.

[Dix and Sas, 10] examined several synergies and opportunities between personal mobile devices and public displays, addressing issues such as the physical size of the situated display, the use and purpose of the mobile devices, the level of integration of the public and personal devices, the movement and physical contact within the interaction, the spatial context of the situated display, and the social context. They analyze two main types of conflict that occur between the interacting users of the public display audience: conflicts of content, and conflicts of pace. Conflicts of content, i.e., what is seen, can occur for various reasons: *"(1) conflict between the use of the screen for displaying content and for displaying interactive feedback (menus, etc.); (2) conflict between different users wanting different specific content (3) conflict between the particular requirements of an individual and maintaining a content stream that is intelligible, useful and engaging for bystanders"*. Conflicts of pace, i.e., when it is seen, includes two types: "*(1) users cannot always have things when they want due to other users requests (c.f. content conflict), the playing of media, etc. (2) users cannot speed-up, slow-down, stop or replay the flow of information because of*

the audience." Resolving these conflicts is a challenge for public displays, particularly for multi-user, multi-application systems.

In the "dual display" approach, Kaviani et al. [Kaviani et al., 09] explore interaction concepts that take advantage of both input and output capabilities of interactive public displays and personal mobile devices. Similarly to Dix et al. [Dix and Sas, 10], they consider two types of conflicts when multiple users attempt to interact and manipulate the same content simultaneously: conflicts in space and conflicts of pace/flow. Conflicts in space mainly originate from the limited screen space to provide visual feedback when executing a sequence of actions or providing information about a new system state. Conflicts of pace/flow usually occur when users do not have any feedback on the system behavior. In order to reduce these conflicts the authors defined four design strategies: localized interactions, distributed system state, providing display focus, and cause summary.

A study by Peltonen et al. describing CityWall [Peltonen et al., 08], a large multi-touch display installed in a central location in Helsinki, Finland, addresses some of these conflicts. The system can visualize images retrieved from Flickr and the users can interact with the content using one- or two-handed gestures, e.g., resize, rotate, etc. This research discusses how people succeed to appropriate a large display, resolve the potential conflicts, and find the right moment to take the turn. Their results show that the decision about when a person should interact does not depend only on the available space at the display, but rather on a set of complex social interactions of reasoning and negotiation between participants. Sacks et al. [Sacks et al., 74] investigate a turn taking system for casual conversation seeking an answer on how do participants in the conversation "select" the next speaker. They build a set of rules that could help organizing the speaker selection. This work is an analogy to ours and we acknowledge its relevance to convey place specific interactions. While social protocols may play an important role in shared use of public displays, in our scenario based on mobile interaction users may not be mutually aware and, therefore, their perception of the control process may have to be entirely grounded on the information provided by the system.

Transversal to public display research, we grounded our approach in a set of recent and emergent trends in Ambient Intelligence such as situated intelligence and global service ecosystem [José et al., 10]. While the first trend enables AmI systems to achieve a close coupling with their social and cultural environment and subsequently provide people a more effective way to act in their environment, the second trend – global services could embed functionality that can be relevant anywhere in an open display environment, thus eliminating the burden to build dedicated services on a case-by-case basis.

3 Selection and Control in GUI Systems

The first phase in our methodology has been to investigate application selection and control mechanisms in traditional GUI systems, more specifically desktop and mobile devices, and analyze to what extent they could be repurposed for public display interaction. As the first step, we selected technical descriptions of application selection and control mechanisms from desktop and mobile interaction GUIs. These were retrieved from various sources, including Wikipedia, books, and various web

sites. We used a total of 31 descriptions referring to 20 different concepts from various interaction models and operating systems (OS), e.g. Windows, UNIX, Mac OS, iOS and Android. A complete listing of the input sources can be found in [Taivan et al., 13]. We then analyzed each description to identify the multiple ways in which applications could be selected and controlled. Each reference to a selection or control technique was coded using open coding and a dedicated coding software. For every code created, a small memo describing its generic meaning was associated. This resulted in 61 codes. At the end, we conducted several consolidation sessions to establish the main concept clusters in relation to application selection and control, resulting in 5 main categories: *Controlling Application Life Cycle, Application Identification, Implicit Application Selection, Explicit Application Selection* and *Visual Layers*. Finally, using these categories and an adapted version of the sensing systems framework by Bellotti et al. [Bellotti et al., 02], we analyze the specificities of application selection and control in public displays. For each category, we review the traditional GUI solutions and analyze the new challenges raised by public displays. When considering those specificities, we assume in particular that there can be many concurrently interacting users in the environment and also that the execution environment of the applications is not a single display, but instead an ecosystem of displays with multiple distributed user interfaces that span across multiple devices, e.g., public displays, mobile phone, touch enabled surfaces.

3.1 Controlling Application Life Cycle.

The application life cycle embodies the sequence of execution states that occur between the launch and termination of an application, e.g., background execution, suspended, inactive, foreground execution. An application can change its state based on users' explicit actions, operating system or application internal events.

For instance, in iOS the application life cycle is composed of five distinct states: *not running* – the state of a rebooted device, *active* – the application is displayed on the screen and receives inputs, *background* – the application may execute code without receiving inputs or update the screen, *suspended* – an application is frozen and its state is stored in RAM and *inactive* – a temporary rest between two other states, e.g., yielded by incoming calls or if the user has locked the screen. While in the active state, an application may require visual and input resources, in the background execution the application is running in a constrained behavior without requiring any display or user input resources.

For public displays, the execution environment should be seen as the physical environment of the displays, where potentially multiple displays may exist. Therefore we should separate application availability in the environment from its presentation on the displays or from its execution on any particular device of that environment. While applications may be expected to be always available and ready to produce content on any display, their normal execution mode may be a waiting mode in which they are ready to receive input signals and in appropriate moments generate content for presentation on the displays. The main challenge is modeling this combination between execution states and presentation state in a way that people can easily perceive and learn to control.

3.2 Application Identification

Application identification is concerned with ways in which users can recognize and distinguish the various apps. Normally, the applications from traditional computing platforms may be identified through icons, a thumbnail photo briefly describing its functionality or, during execution, by using specific system level indicators of common app description fields, e.g., window title, favicon. For instance, in Windows and Mac desktop environments there is a system-based software application that lets users identify and inspect all applications in execution and their respective processes or tasks. In particular, in Mac OS, the name of the foreground app is always on the menu bar. However, in mobile devices, many running applications do not have a clear application title.

Application icons allow users to easily recognize and launch applications. It is represented as a small picture, which intuitively describes the function of the respective program. An application icon is designed to be language independent (does not contain any text) and it offers rapid entries in the system functionalities. Application icons may also be extended to present key application state. For example, Windows 8 features a user interface paradigm based on the concept of live tiles that are dynamic icons with a larger size that identify the respective app and shows app specific data at the same time, e.g., the number of unread email messages in a mail application.

For public displays, identifying applications is also important so that people may associate the content they see on the displays with the application generating that content. An adapted version of GUI concepts, such as application titles may be used in some cases, but may also be inappropriate in other cases because it may interfere with the rich visualization requirements of public displays. Alternative approaches may include a list of the applications that are currently available to be shown on the displays. This list may include the application id and a summary of its content, e.g. live tile, and may be available through mobile devices or occasionally shown on the display to prompt interaction.

3.3 Implicit Application Selection

Implicit Application Selection is initiated automatically by the system or by the application themselves. The system-based activation is an additional way to launch applications as a result of various event triggers. This may include time-based events or certain system events, e.g. a device join or a change in network availability. When an event occurs it may trigger background or foreground application activation. In Windows OS, as a consequence of system-based scheduling, most processes are launched in background mode without any user interface, e.g. Server, Network Connections. The application based selection may also entail application specific logic that triggers its appearance, e.g., from background to foreground. This approach is very common in mobile devices where various applications can be triggered by external events, e.g. a phone call.

For public displays, implicit application selection may be part of a regular scheduling process in which the systems iterates over the multiple applications available, but it may also triggered by external events. In a mixed-initiative model, the system would need to implicitly call for specific apps, even if there is no activity from

users. Additionally, some applications may only be relevant when particular contextual conditions occur. In such case the system may at any moment make selections based on the interpretation of the respective context, e.g. people presence and their preferences. Therefore, a challenge for pervasive displays is the ability to integrate this dynamic application selection into the application execution mode.

3.4 Explicit Application Selection

Application selection is an action in which a user requests the activation of a particular application. This may correspond to the initiation of the application or to a change in its execution state, e.g., from background to foreground. Selecting applications from the whole list of available applications can be complex because of the potentially large number of applications. Most operating system offer some type of short-list of commonly used applications, either as more specific sub-lists or through particular GUI elements, e.g. application icons on a desktop environment. Selecting from a list of active applications is contextually very relevant and can be accomplished through specific tools, such as the taskbar, app switch shortcuts, or app docks. For example, in Windows OS, Mac, KDE, and UNIX a specific key combination, i.e., Alt+Tab, switches foreground execution between the most recent top-level application windows.

For public displays, viewers need firstly to be able to identify available applications and request them. The first challenge is which activation techniques can be more adequate for public displays. Subsequently, given the multi-user context of public displays, another challenge is the mediation between possible conflicting requests from multiple users or even between users and system goals.

3.5 Visual Layers

Visual layers enable multiple applications to be simultaneously active while sharing a single visual display. The existence of a single foreground layer coordinates where the current focus is and therefore to which application system input should be directed. Applications may trigger visual attention by using a special, always-on-top layer. This is often used for splash screens or in other cases to force users to attend an interaction request. A similar goal can also be achieved through other particular types of visual layer: notifications. In a traditional OS, a notification message warns users about application data updates or about system level issues. Mainly, the computer notifications contain two classes: a) one that calls for user attention, e.g., pop-ups and b) the other that does not call for explicit user attention, e.g., pop-under. A pop-under notification contains a non-intrusive content that resides behind scene. In Windows environments, non-intrusive notifications are shown in the notification area situated in the right side of the Taskbar.

For public displays, multiple visual layers can also be an important feature. However, considering that multiple people may be sharing the display, it becomes much more challenging to achieve a balanced combination between multiple layers and a good interaction experience. Still, well-designed notification layers that choose the best time to present themselves may provide an important alternative channel for presenting contextually relevant content outside the normal presentation cycles of the applications. In particular, these alternative visual layers may be important in

generating feedback for users trying to interact with the system and support progressive interaction modes in which users and displays are increasingly aligned while minimizing interactions by accident, such as in gesture-based interfaces.

3.6 Analysis

These techniques from traditional GUI systems provide a worthy starting point for considering specific solutions for public displays and particularly for grounding a systematic analysis of the challenges of application selection and control in public displays. Inspired by the approach described by Bellotti et al. [Bellotti et al., 02] for analyzing the challenges of sensing systems, we also analyze the specificities of application selection and control in public displays by explicitly stating the traditional approaches alongside the new challenges associated with this specific problem domain, as described in [Table 1]. These may also be formulated as five questions that a public display designer should be able to answer to support appropriate application selection and control mechanisms.

Basic Questions	Traditional GUI Concepts	Challenges for public displays
How can the system or the people using it control the application life cycle?	Users click on buttons; System priorities; Triggers; Execution as service; Execution states.	How to model the combination between execution states and presentation state, in a way that people can easily perceive and learn to control?
How do viewers identify applications?	Static and dynamic icons, e.g., live tiles, windows title bar, favicon, etc.	How to raise awareness about installed or running applications? How to associate content being presented with the application? How to address an application?
How does the system implicitly select applications?	System events, e.g., device join or network availability	How to integrate the dynamic application selection into the default application execution mode? How to select relevant apps based on current context?
How do viewers explicitly select applications?	Keyboard, mouse. Start button, Start menu, App Switch (Alt+Tab), app dock, folders or taskbar.	Which interaction techniques make sense in public displays? How to mediate between possible conflicting requests from users?
How to use multiple layers to enhance application selection and control?	Applications windows and notifications; system level notifications, e.g., pop-up, pop-under.	How to effectively use multiple visual layers without disturbing the current content presentation and overall user experience?

Table 1: GUI solutions and public display challenges for application selection and control

4 Field Study Description

In order to explore the issues involved in application selection by concurrent users, we deployed a multi-application display system in a public setting for assessing the multi-user experience of influencing the sequence of applications. Our experiment involved two main dimensions: 1) observing users' behavior and comments, e.g. engagement, embarrassment, frustration, concentration, enthusiasm and 2) assess the responsiveness of the system logic.

4.1 Experimental setup

The experiment involved the deployment of a single public display in a bar at our University Campus. The display was able to show a particular application when requested by users. In our display deployment we used Instant Places platform [José et al., 12] as the underlying technology for creating and managing applications and also for supporting user interactions. We also used four applications from Instant Places, more specifically: 1) *Posters* app that presents posters created by the community and which were approved by display owner; 2) *Football* app that reflects the users' football preferences when they check-in to the respective venue; 3) *Presences* app that lists the nicknames and the peoples' profile images who recently checked-in in the place and 4) *Place Stream* app that presents the history of events generated by the users within the place in which the display is situated [Figure 1].

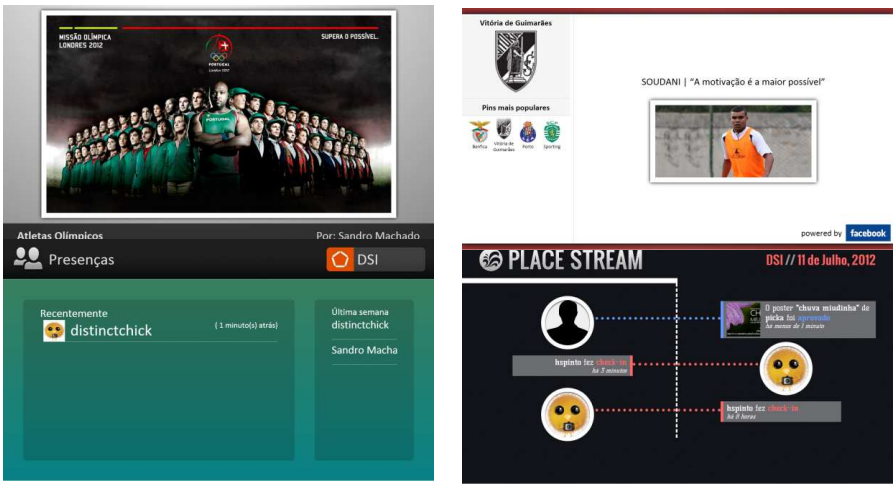

Figure 1: Instant Places applications that were used for the selection process

These four applications shown in full screen were conceived outside the scope of this paper [José et al., 12] and we just use them for application selection and control purposes. Within our experiment we concentrated on how multiple users succeeded to select an application rather than interact with it, e.g., people were not allowed to publish new posters. They just tested the application selection mechanism by

requesting an application to be displayed earlier than waiting for it to be shown as part of the implicit or default presentation list.

To present the applications for this experiment, we developed a specific player with the distinctive ability to combine implicit application selection with the explicit requests. While the display system was presenting the four applications in a time based schedule with 1 minute time slot as a part of the implicit behavior, the requested applications were shown as part of the explicit presentation behavior – starting with the highest requested to the least requested app. To provide users with feedback on the number of requests, we implemented a display web app called ShowRequests that had 1 minute time slot to inform users about the ranking of the preferred apps. Based on this feedback participants could understand how the decisions of applications presentation were taken. The ShowRequests app was presented right after the default behavior sequence and before the explicit application sequence. For instance, one who selected an app in the first part – default sequence, can have the app displayed right after the ShowRequests app. In this way, participants could have their preferred apps presented earlier [Figure 2].

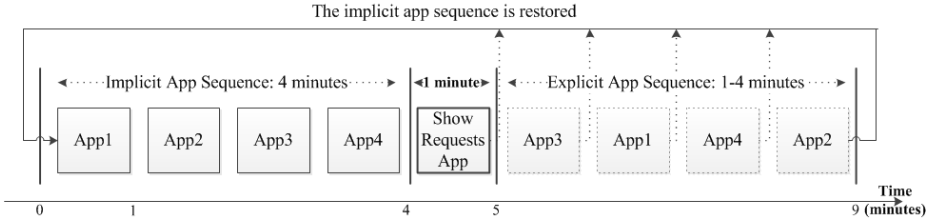

Figure 2: Applications presentation logic

To issue application requests, participants used a mobile web app – *MakeRequests*. They were allowed to request at least one app. The mobile app simply showed four icons of the applications that were running on the large displays. An application request was issued by touching or clicking one of those icons. A confirmation pop-up with the server request acknowledgment was raised after each selected app.

Table [Table 2] summarizes the set of techniques that we have implemented in our experimental system to address the various application selection and control challenges that were identified in [Section 3].

4.2 Evaluation

For the evaluation of the proposed techniques, we invited a group of 6 participants to engage in a situation of concurrent application selection in a real setting. Participants were master students and researchers that had no previous knowledge about the display system or the logic behind applications selection. In our experiment we concentrated only on applications selection. Therefore, the value of the applications or the users' interactions with them was not addressed in this study.

Basic Questions	Public Displays Techniques
How can the system or the people using it control the application life cycle?	Time based invocation of the newly selected apps. Applications have just one execution state, i.e., full screen content presentation.
How do viewers identify applications?	Splash screens; App icons in ShowRequests and MakeRequests apps.
How does the system implicitly select applications?	Fixed timeline with 1 minute time slot; The implicit applications sequence is repeated after dynamically allocated time slots of users' preferred apps.
How do viewers explicitly select applications?	Selection in mobile application; Voting logic to deal with conflicts.
How to use multiple layers to enhance application selection and control?	Notification messages in users' mobile devices; Full screen display app with the ranking of the requested apps, i.e., ShowRequests app.

Table 2: Application selection and control techniques in public displays

We briefly introduced participants into the applications' functionalities and invited them to make various application selections by using the MakeRequests mobile app. We chose not to provide any details on the system selection process or how the system combines implicit selection with explicit selection from potentially multiple users. This way, the experiment could be closer to a real world situation, where people would not normally have the time or willingness for lessons about public displays behavior. People had approximately half an hour to test how the system worked. In the end, we conducted a semi-structured interview (5-7 minutes) in which we asked participants about the experience with the system. The interviews were audio recorded and transcribed in text. Overall, the experiment had two main data sources: 1) live observation of users' behavior while selecting applications and 2) users' opinions about the application selection and control logic.

4.3 Results

Understanding the application presentation logic and making a clear distinction between the implicit and explicit applications presentation sequences was not so easy. In the beginning every participant requested at least one application and many requests were concurrently issued. This has considerably increased the complexity of the overall selection process and made it much more difficult for each individual participant to understand how the system was taking into account his or her requests. After roughly 15 minutes, participants started to focus more on making sense of system behavior rather than just issuing requests. They started to pay more attention to the notification messages provided by ShowRequests public display app. At a certain point, they agreed to make a combined vote that would enable them to more easily uncover the application selection logic. They all voted for the *Presences* app. In this way, they understood how the system resolved the requests and succeeded to familiarize with our approach for influencing the applications turn in a large display.

> *„In the beginning all of us selected all the apps and that's why it was difficult to clearly uncover its functionality."*

Participants remarked that the logic was simple, although it required close observation to figure out the content presentation order. The logic was categorized as fair by all the people involved and there was no lock on a certain application due to the implicit presentation behavior.

> *„What I know is, it took me a while to see what the system is and it takes really close observations".*
> *„The application selection logic is reasonably fair for everyone".*

Participants noted that the system had low responsiveness and they would like to have more immediacy in showing the apps. For instance, given the timeline of implicit application behavior of 4 minutes, one could get his preferred app on screen after those 4 minutes for the best case, and 7 minutes for the worst case. This fact highlighted that implicit application presentation behavior could not be interrupted thus imposing a minimum delay of 4 minutes. Users clearly found this delay a limitation and a sign of low responsiveness. They expected more immediacy for presenting the requested apps, e.g., allow them to interrupt the default presentation order or schedule and get the apps displayed within 1 minute time span – for the worst case. Furthermore, for the case of many applications the solution was categorized as not being scalable – a lowest requested app would be shown most of the time in the end.

> *"… I believe people want the requested apps to be presented more immediately."*

Although our display system employed two types of notifications, participants gave us a lot of comments about the need for rich notifications, both for public display and mobile devices. They liked to be informed about when their particular requests were served, when was the appropriate time to vote or when the requested apps were being selected by the system. This importance of the feedback in the interaction process is in line with previous findings by Kaviani et al. [Kaviani et al., 09].

> *"You can use more instructions in the MakeRequests mobile web app. You do not use the screen. The screen can be used for more information, e.g., nr. of requests. Would be better to inform people when a particular requested app will be shown and when is the default or implicit behavior."*

Participants further suggested that a well-designed set of notifications could keep users engaged without losing their interest for the desired application. For example, a good system improvement could be a status bar for both devices (public and personal displays) presenting what the next app is, which is the current applications sequence and when people can request apps. Moreover, the MakeRequests mobile app could have the ranking of the requested apps and even a history of the previously most selected apps.

> *"While we are using the MakeRequests mobile app, we may have more information, e.g., history of the previously requested apps. Also, would be nice to inform people when the time to vote is."*

Other notifications recommended for mobile clients were those that informed users about the presentation time of a requested application (e.g., "*Watch the screen, this is your preferred application*") or when exactly that will happen (e.g., "*In less than 1 minute your requested app will be shown*").

4.4 Discussion

Our display system deployment focused on the evaluation of a set of techniques involved in the process of application selection and control. An appropriate design for application selection and control techniques is one that delivers acceptable responsiveness and rich and clear notifications in such a way that people may reason about system behavior and how they can affect it. In the following we are discussing three design considerations we reached as a result of our system deployment.

4.4.1 Making sense of the system behavior

Our multi-application display system presented applications based on a mixed initiative model in which the system and users influenced the applications presentation sequence. Our applications had a simple lifecycle with just one execution state, i.e., full screen content presentation. A distinct use case was mentioned by a user who would like to "*freeze*" an application presentation in order to get its entire content digested. We believe that there are various similar scenarios in which the existence of a rich application lifecycle would be a strong requirement for future display systems. For instance, the *Presences* app might go from full screen presentation state to a background execution state from which it can pop up unobtrusive data notifications and release the full screen side for other applications that really needs more screen space, e.g., *Poster* app.

4.4.2 Alignment

In our experiment, participants had difficulties in monitoring the system response to their actions. We only provided the minimal notifications mechanisms both for large display and mobile devices. For instance, our four applications could be identified based on their titles and associated splash screens. While the *Presence* and *Place Stream* apps could be easily identified by their titles in the top side of the screen, *Poster* and *Football* apps had rich visualization requirements and only employed splash screens as a technique to avoid information overloading and still deliver high resolution content.

The mobile app somehow created an analogy of what is presented in the large display, though it had a simple interface of showing the application icons and titles. The clue here was that our web apps (display and mobile) used the same data source or API to get the application metadata and implicitly the graphical representation in the mobile devices was similar with the implicit presentation sequence in the large display.

Given the rich application lifecycle requirements of public display environments – as depicted in [Section 3], we learnt that people require various ways for app identification. The issue can become very complex if we consider that applications may be run in background mode. Still, even for our case, the simple and single visual layers we deployed were not sufficient and there is a need for a set of GUI elements and concepts that could complement each other and better characterize this particular interaction scenario, e.g., status bar with application names or rich information user interfaces for the mobile clients. One user clearly stated that our display system required "*close observation*" and this was because of the lack of visual layers we implemented.

The need for more notifications for both public display and mobile clients was also motivated by the real life scenario where one can be distracted and lose the notification delivered by the public display. To overcome this issue, participants suggested the implementation of similar notifications for the mobile clients, e.g., checking the applications requests on mobile devices. Whilst achieving a balanced combination between multiple layers and a good interaction experience constitute apparently a non-complex issue, our observations uncovered the opposite. Despite the fact we had only one full screen application presentation at the time, which meant simple display layout, people wanted to get personal notifications about the effect their commands would have on the current application presentation.

4.4.3 Fairness and responsiveness

Participants found difficult to grasp the overall content presentation experience given the existence of the explicit application presentation sequence. People often mentioned that "*The system does not seem to respond*". Therefore, the interviews reflected the need for a system that presents the users' selected applications in a more responsive way, i.e., interrupt the default behavior and have the requested applications presented in less than 4 minutes. Others believed that 1 minute could be a long time presentation for certain applications, e.g., *Presences* app.

Even so, the application selection logic appeared to be fair for all the users involved and enabled a systematic approach in resolving multi-user requests as long as people understood the system and set their expectations accordingly. Firstly, this means that participants' expectations had an important role to set their perceptions about the fairness and responsiveness of our system logic – a fact that we clearly observed and stated throughout the interviews. For example, one user said that could not find any other fair solution in that moment to improve the way applications are presented and still avoid the application locks.

The system responsiveness was clearly seen as a limitation, but user' expectations were very important in regard to this issue. We never conceived the system to respond immediately and in most realistic scenarios, people would not get an immediate response from a display. However, proper feedback has shown to be crucial in leading people to understand this and accept that their request was being given attention and would be answered soon. If that was the case, perhaps responsiveness has been understood less negatively.

5 Conclusions

Future open display system will tend to increase their value proposition by allowing content from multiple sources and concurrent applications that are able to continuously react to multiple viewers. Such open display environments are facing radically new challenges for application selection and control.

In this paper we investigated the challenges of designing appropriate techniques for application selection and control in multi-application and multi-user display systems. Getting inspiration from traditional GUI systems, we came up with a systematic approach for identification and aggregation of challenges involved in the design of novel applications control mechanisms. In order to address the identified challenges, we designed, implemented and evaluated a set of techniques for application selection and control. This was handled as part of an experiment in which we deployed a public display system that supported multiple applications and which was able to receive explicit application presentation requests from multiple viewers.

Our results indicated some success in enabling viewers with the necessary techniques to influence the application sequence in a public display and to mediate the potential conflicting requests. While all of the participants reported the easiness and fairness of the application selection approach, they also noticed the need for more responsiveness and rich notifications for the various human-computer interaction stages, both for public and private devices. Further research may consider an improved version of the respective application control mechanisms and a long-term deployment based on real world scenarios.

Acknowledgements

The research has received funding from the European Union Seventh Framework Programme (FP7/2007-2013) under grant agreement no. 244011 (PD-NET) and from "Fundação para a Ciência e a Tecnologia", under the research grant SFRH/BD/75868/2011. We would like to thank Samuel Moreira for his comments about the display system deployment, Ana Melro for helping us in organizing the experiment and all the people involved in this study.

References

[Bellotti et al., 02] Bellotti, Victoria, Maribeth Back, W. Keith Edwards, Rebecca E. Grinter, Austin Henderson, and Cristina Lopes. 2002. "Making Sense of Sensing Systems." In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems Changing Our World, Changing Ourselves - CHI '02, 415. New York, New York, USA: ACM Press. doi:10.1145/503376.503450. http://dl.acm.org/citation.cfm?id=503376.503450.

[Boring et al., 13] Boring, Sebastian, and Dominikus Baur. 2013. "Making Public Displays Interactive Everywhere." IEEE Computer Graphics and Applications 33 (2) (March): 28–36. doi:10.1109/MCG.2012.127. http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=6381421.

[Davies et al., 09] Davies, Nigel, Adrian Friday, Peter Newman, Sarah Rutlidge, and Oliver Storz. 2009. "Using Bluetooth Device Names to Support Interaction in Smart Environments." In Proceedings of the 7th International Conference on Mobile Systems, Applications, and

Services - Mobisys ’09, 151. New York, New York, USA: ACM Press. doi:10.1145/1555816. 1555832. http://portal.acm.org/citation.cfm?doid=1555816.1555832.

[Davies et al.,12] Davies, Nigel, Marc Langheinrich, Rui José, and Albrecht Schmidt. 2012. “Open Display Networks: A Communications Medium for the 21st Century.” IEEE Computer 45 (5) (March 14): 58–64. http://www.computer.org/portal/web/csdl/doi/10.1109/ MC.2012. 114.

[Dix and Sas, 10] Dix, Alan, and Corina Sas. 2010. “Mobile Personal Devices Meet Situated Public Displays: Synergies and Opportunities.” IJUC: 11–28.

[Hosio et al., 10] Hosio, Simo, Marko Jurmu, Hannu Kukka, Jukka Riekki, and Timo Ojala. 2010. “Supporting Distributed Private and Public User Interfaces in Urban Environments.” In Proc. 11th Workshop on Mobile Computing Systems and Applications (HotMobile 2010), Annapolis, MD, USA, 25-30, 25–30. New York, NY, USA: ACM. doi:10.1145/1734583. 1734590. http://doi.acm.org/10.1145/1734583.1734590.

[Izadi et al., 03] Izadi, Shahram, Harry Brignull, Tom Rodden, Yvonne Rogers, and Mia Underwood. 2003. “Dynamo: a Public Interactive Surface Supporting the Cooperative Sharing and Exchange of Media.” In Proceedings of the 16th Annual ACM Symposium on User Interface Software and Technology - UIST ’03, 159–168. New York, New York, USA: ACM Press. doi:10.1145/964696.964714.
http://dl.acm.org/citation.cfm?id=964696.964714.

[José et al., 12] José, Rui, Hélder Pinto, Bruno Silva, Ana Melro, and Helena Rodrigues. 2012. “Beyond Interaction: Tools and Practices for Situated Publication in Display Networks.” In Proceedings of the 2012 International Symposium on Pervasive Displays - PerDis ’12, 1–6. New York, New York, USA: ACM Press.
http://dl.acm.org/citation.cfm?id=2307798.2307806.

[José et al., 10] José, Rui, Helena Rodrigues, and Nuno Otero. 2010. “Ambient Intelligence: Beyond the Inspiring Vision.” JUCS.
http://www.jucs.org/jucs_16_12/ ambient_intelligence_ beyond_the.

[Kaviani et al., 09] Kaviani, Nima, Matthias Finke, Sidney Fels, Rodger Lea, and Hua Wang. 2009. “What Goes Where?” In Proceedings of the First International Conference on Internet Multimedia Computing and Service - ICIMCS ’09, 129. New York, New York, USA: ACM Press.
doi:10.1145/1734605.1734637. http://dl.acm.org/citation.cfm?id=1734605.1734637.

[Kostakos et al., 13] Kostakos, Vassilis, Hannu Kukka, Jorge Goncalves, Nikolaos Tselios, and Timo Ojala. 2013. “Multipurpose Public Displays: How Shortcut Menus Affect Usage.” IEEE Computer Graphics and Applications 33 (2) (March): 56–63.
doi:10.1109/MCG.2012.125.http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=637366 5.

[Kubitza et al., 12] Kubitza, Thomas, Sarah Clinch, Nigel Davies, and Marc Langheinrich. 2012. “Using Mobile Devices to Personalize Pervasive Displays.” ACM SIGMOBILE Mobile Computing and Communications Review 16 (4) (February 4): 26–27.
http://dl.acm.org/citation.cfm?id=2436196.2436211.

[Morales-Aranda and Mayora-Ibarra, 07] Morales-Aranda, A.H., and O. Mayora-Ibarra. 2007. “A Context Sensitive Public Display for Adaptive Multi-User Information Visualization.” In Third International Conference on Autonomic and Autonomous Systems (ICAS’07), 63. doi:10.1109/CONIELECOMP.2007.40. http://ieeexplore.ieee.org/stampPDF/ getPDF.jsp?tp=& arnumber=4437940.

[Ojala et al., 12] Ojala, Timo, Vassilis Kostakos, Hannu Kukka, Tommi Heikkinen, Tomas Linden, Marko Jurmu, Simo Hosio, Fabio Kruger, and D Zanni. 2012. "Multipurpose Interactive Public Displays in the Wild: Three Years Later." IEEE Computer, May.

[Pablo et al., 06] Pablo, Haya, Montoro Germán, Abraham Esquivel, Manuel García-Herranz, and Alamán Xavier. 2006. "Mechanism for Solving Conflicts in Ambient Intelligent Environments." JUCS. http://www.jucs.org/jucs_12_3/a_mechanism_for_solving/ jucs_12_03_ 0284_0296_haya.html.

[Peltonen et al., 08] Peltonen, Peter, Esko Kurvinen, and Antti Salovaara. 2008. "It's Mine, Don't Touch!: Interactions at a Large Multi-touch Display in a City Centre." In Proceedings of CHI2008, 1285–1294. http://dl.acm.org/citation.cfm?id=1357255.

[Sacks et al., 74] Sacks, Harvey, Emanuel A Schegloff, and Gail Jefferson. 1974. "A Simplest Systematics for the Organization of Turn-Taking for Conversation." Language 50 (4): 696–735. doi:10.1121/1.1905050. http://www.jstor.org/stable/412243.

[Taivan et al., 12] Taivan, Constantin, Rui José, and Ivan Elhart. 2012. "Selection and Control of Applications in Pervasive Displays." In 6th International Conference on Ubiquitous Computing & Ambient Intelligence. Victoria-Gasteiz, Spain: Springer LNCS.

[Taivan et al., 13] Taivan, Constantin, Rui José, Bruno Silva, Ivan Elhart, and Jorge Cardoso. 2013. "Design Considerations for Application Selection and Control in Multi-user Public Displays - Coded Text Fragments."doi:10.6084/m9.figshare.687168. http://dx.doi.org/10.6084/ m9.figshare.687168.

[Terrenghi et al., 09] Terrenghi, Lucia, Aaron Quigley, and Alan Dix. 2009. "A Taxonomy for and Analysis of Multi-person-display Ecosystems." Journal of Personal and Ubiquitous Computing 13 (8): 583 – 598.

[Vogel and Balakrishnan, 04] Vogel, Daniel, and Ravin Balakrishnan. 2004. "Interactive Public Ambient Displays: Transitioning from Implicit to Explicit, Public to Personal, Interaction with Multiple Users." In Proceedings of the 17th Annual ACM Symposium on User Interface Software and Technology - UIST '04, 137. New York, New York, USA: ACM Press. doi:10.1145/1029632.1029656. http://dl.acm.org/citation.cfm?id=1029632.1029656.