P-LAYERS – A layered framework addressing the multi-faceted issues facing community-supporting public display deployments

NEMANJA MEMAROVIC, University of Lugano
MARC LANGHEINRICH, University of Lugano
KEITH CHEVERST, Lancaster University
NICK TAYLOR, Newcastle University
FLORIAN ALT, University of Stuttgart

The proliferation of digital signage systems has prompted a wealth of research that attempts to use public displays for more than just advertisement or transport schedules, such as their use for supporting communities. However, deploying and maintaining display systems "in the wild" that can support communities is challenging. Based on the authors' experiences in designing and fielding a diverse range of community-supporting public display deployments, we identify a large set of challenges and issues that researchers working in this area are likely to encounter. Grouping them into five distinct layers – (1) hardware, (2) system architecture, (3) content, (4) system interaction, and (5) community interaction design – we draw up the P-LAYERS framework to enable a more systematic appreciation of the diverse range of issues associated with the development, the deployment, and the maintenance of such systems. Using three of our own deployments as illustrative examples, we will describe both our experiences within each individual layer, as well as point out interactions between the layers. We believe our framework provides a valuable aid for researchers looking to work in this space, alerting them to the issues they are likely to encounter during their deployments, and help them plan accordingly.

Categories and Subject Descriptors: **H.4.3**. Communications Applications; **H.5.2. [User Interfaces]**: User-centered design; **H.5.3. [Group and Organization Interfaces]**: Theory and Models; **J.4. [Social and behavioral sciences]**: Sociology

General Terms: Design

Additional Key Words and Phrases: Community interaction, public displays, pervasive displays, community needs

ACM Reference Format:

Memarovic, N., Langheinrich, M., Cheverst, K., Taylor, N., and, Alt, F. 2013. P-LAYERS – A layered framework addressing the multi-faceted issues facing community-supporting public display deployments. ACM Trans. Compt.-Hum. Interact. XX, YY, Article ZZ (March 2013), 31 pages.
DOI:http://dx.doi.org/10.1145/0000000.0000000

The research leading to these results has received funding from the European Union Seventh Framework Programme (FP7/2007-2013) under grant agreement no. 244011, the EPSRC-funded EPSRC CASIDE (grant number EP/C005589/1) and SHARC (grant number EP/K015850/1) projects and a Microsoft Research European PhD scholarship.

Author's addresses: N. Memarovic and M. Langheinrich, Faculty of Informatics, University of Lugano, Via Giuseppe Buffi 13, 6900 Lugano, Switzerland; K. Cheverst, School of Computing and Communications, Lancaster University; N. Taylor, Culture Lab, School of Computing Science, Newcastle University; F. Alt, Institute for Visualization and Interactive Systems, University of Stuttgart.

Transactions on Computer Human Interactions

DOI:http://dx.doi.org/10.1145/0000000.0000000

1 INTRODUCTION

Today's public spaces see an increasing deployment of digital displays: they list interesting facts and events at universities, display schedules and news in metro stations, present special offers in shopping malls, or advertise a product on a building facade. Yet their predominant use as simple slide presenter and video player has seen dwindling "eyeballs" and led to *display blindness* [Huang et al. 2008; Müller et al. 2009] – an effect where viewers ignore much, if not most, of such animated advertisements. Researchers have started to suggest a range of alternative use cases for public displays: they can allow locals to share historical photos of a place [Taylor and Cheverst 2009] (discussed in detail in section 2), display the logos of football clubs that coffee-shop patrons are supporting [José et al. 2012], or summarize the interests of people in the vicinity [McCarty et al. 2001; McDonald et al. 2008; McCarthy et al. 2009]. In all of these examples, public display technology is used to convey a sense of community to the display's viewers by stimulating interaction with, and awareness of, other community members. In this way, public displays help to enrich the social functions of public spaces, which provide a place where people can socialize, relax, and learn something new – ultimately creating emotional connections with others [Carr et al. 1992].

The design [Memarovic et al. 2012a], deployment [Ojala et al. 2011] and evaluation [Cheverst et al. 2008] of public display systems to support community interaction is challenging. Ultimately, the goal is to stimulate some form of *community interaction*. This can be as simple as encouraging people in the display's vicinity to talk to each other [Memarovic et al. 2012b] or, more indirectly, by allowing community membership to be expressed in some form, e.g., through badges [José et al. 2012]. Displays can be used to explicitly exchange information among community members [Churchill et al. 2003; Redhead and Brereton 2009; Taylor and Cheverst 2009; Alt et al. 2011b] (Alt et al. 2011b will be discussed more in section 4) or to prompt passers-by to play for their community in a competitive game running on the display [Memarovic et al. 2011b]. These different types of interventions typically require different *system interaction* capabilities. Some need active touch-screen input; others work with short-range wireless communication devices, such as Bluetooth-enabled phones. Displays might be located outdoors in busy town centers or inside quiet village cafes. These interaction choices, in turn, have a strong impact on the type of *content* that is needed and/or supported. In some cases, content can be contributed by community members (e.g., classified advertisements on a bulletin board); in other cases, editorial content can be more suitable (e.g., questions for a trivia quiz game). Depending on both the source of content and the envisioned interaction with it, different *system architectures* are needed. Some interventions might require cross-device access (e.g., accessing classifieds from a website or mobile phone [Alt et al. 2011b]) while others need to support content caching to cope with disconnection problems [Taylor and Cheverst 2009; Memarovic et al. 2011b]. Last but not least, appropriate *hardware* is required. In some cases it is possible (or even necessary) for researchers to introduce their own customized hardware into a setting (e.g., a custom installation in a bus underpass [Clinch et al. 2011]) while other deployments can (or must) use pre-installed hardware (e.g., an existing display network in a city [Alt et al. 2011b; Memarovic et al. 2011b]).

The five above-mentioned factors – *community interaction design, system interaction, content, system architecture,* and *hardware* – can be arranged in a layered fashion (cf. Fig. 13) to illustrate the dependencies between them, as well as their constructive structure in the context of community-building public display deployments.

The factors – and the interplay between them – emerged from our own experiences designing, developing, and evaluating public display systems "in the wild" that supported communities [Taylor and Cheverst 2009; Alt et al. 2011b; Memarovic et al. 2011b]. We believe that these three deployments – Wray, FunSquare, and Digifieds, form a representative set of systems for supporting community interaction.

Their layered arrangement, together with a set of analysis methods (described in section 5) form a framework that can be used both before and during a community-supporting public display deployment in order to allocate resources, uncover hidden issues, and troubleshoot emerging problems. We call this the P-LAYERS framework (from "Public display LAYERS" and pronounced "players"). Our contribution is thus twofold:

- We describe in section 5 the P-LAYERS framework that provides a *layered overview of the challenges* that *researchers* face when building and deploying public display systems "in the wild" that support community interaction. The vertical arrangement of the layers illustrates that in order to reach community interaction research teams of public display applications need to build on top of a number of supportive layers.
- We discuss in section 6 how the P-LAYERS framework can be used to support research teams with resource allocation and deployment troubleshooting. The framework can be used to (1) calibrate one's individual awareness of the various issues involved in successful community deployments, given one's interests (section 6.1). By analogy, it is akin to viewing the various layers as a half-hidden iceberg, with the waterline being one's preliminary research interests. It can also be used to (2) tabulate encountered issues during a deployment in order to uncover follow-up issues (section 6.2). Also (3) different shapes of the framework can be used to better judge the necessary effort for each layer (section 6.3).

In the remainder of this introductory section we describe our orientation to the terms "community" and "community interaction", summarize related work on public displays that stimulate community interaction, and describe two settings in which we deployed public display applications for stimulating community interaction. We then present three sections that each describe our experiences in developing, deploying and evaluating such an application: the Wray Photo Display [Taylor and Cheverst 2009; Taylor and Cheverst 2012] in section 2, FunSquare [Memarovic et al. 2011b, Memarovic et al. 2012b] in section 3, and Digifieds [Alt et al. 2011b, Alt et al. 2013] in section 4. Section 5 presents the P-LAYERS framework and its individual layers in detail along with illustrative cross-cutting examples of the interplay between the layers. We then discuss the use of the framework to plan, reflect, and troubleshoot community-supporting public display deployments in section 6, before closing with concluding remarks and future work in section 7.

An alternative to reading the sections in strict sequence is to skip directly to section 5 to read about the framework first, and then backtrack to sections 2 - 4 to learn about individual systems that informed the design of the framework. Concrete advice on how to apply the framework can be found in section 6.

1.1 Definitions of Community and Community Interaction

There are many definitions of "community". In 1955, Hillery pointed out no less than ninety-four different definitions [Hillery 1955] while, more recently, Clark noted the continuing change in the meaning of the term [Clark 2007]. One reason for this diversity can be attributed to the different types of communities that exist. For example, *communities of practice*, as defined by Wenger, refers to groups of people tied through a common craft or profession [Wenger 1998]. Alternatively, *communities of interest,* according to Fischer describes groups of people who have a common interest in a topic [Fischer 2001], while *place-based communities, as* defined by Ramsey and Beesley, relates to groups of people that reside and thrive within a geographical location [Ramsey and Beesley 2007]. In the context of our work on using public displays for stimulating community interaction, we have mainly focused on place-based communities.

It is generally acknowledged that one of the core ingredients of any community is that it conveys a shared sense of belonging, a *sense of community*. According to McMillan and Chavis, this sense of community originates from four main factors [McMillan and Chavis 1986]: 1) membership, 2) influence, 3) integration, and 4) shared emotional connection. *Membership* reflects one's notion of belonging to a community. *Influence* refers to the ability of a member to make a change and impact upon the community and vice versa. *Integration* relates to the reinforcement of community ties over time. *Shared emotional connection* refers to having a shared notion of the community meaning and its values among the members.

A sense of community can increase through *community interaction*, i.e., interaction and exchange that happens between members of a community. Community interaction can take various forms: it can be face-to-face interaction (e.g., direct social interaction between people); knowledge or material exchange (e.g., through advertisement on a local bulletin board); contributing to building a shared history (e.g., contributing an image to a community album); or taking part in joint community causes (e.g., promoting the community and its values or simply taking part in a community organized event). While community interaction can also go beyond a single, physically localized community (see, e.g., [Mynatt et al. 1997], [Clark 2007], [Memarovic and Langheinrich 2010], and [Memarovic et al. 2011a] as examples), our focus on place-based communities implies interaction processes involving local community members.

1.2 Situated Displays and Communities

Previous research has explored the use of public displays for community interaction in a variety of specific settings, namely: urban areas, "third places" as cafés, working environments, and rural places.

In the urban environment, Peltonen et al. deployed the CityWall [Peltonen et al. 2008], a large multi-touch display located in the city center of Helsinki, Finland. This installation displayed randomly chosen Flickr images tagged with 'Helsinki', which multiple users could browse in a playful manner. Cityspeak [Lévesque et al. 2006] and Digifieds [Alt et al. 2011b] were similarly deployed in an urban context, yet allowed users to post their own content to the public display. The Wray Photo Display [Taylor and Cheverst 2009; Taylor and Cheverst 2012], Story Bank [Jones et al. 2008], Nnub [Redhead and Brereton 2009] and BigBoard [Maunder et al. 2011] are examples of public display-based community applications that were deployed in rural environments [Jones et al. 2008; Taylor and Cheverst 2009] and local neighborhoods [Redhead and Brereton 2009; Maunder et al. 2011]. The Wray Photo Display application showed community-sourced images, e.g., from festivals and historical events. Similarly, Story Bank allowed villagers in a rural community in India to share stories in the form of images and audio files. Nnub allowed people to post classifieds in a similar fashion as Digifieds [Alt et al. 2011b] yet with a more local focus on a single neighborhood while BigBoard supported no-cost media sharing for users at a 'Learn to Earn' facility in a township close to Cape Town, South Africa.. Deployments in working environments range from displaying personal interests on a screen [McCarty et al. 2001; McDonald et al. 2008; McCarthy et al. 2009], sharing images, stories and Web links [Churchill et al. 2003], to displaying community news in the form of a newspaper [Houde et al. 1998]. Finally, examples of display deployments to support community interaction in third places showed information from online profiles and presence information [McCarthy et al. 2009], user generated content in form of pins and poster (José et al. 2012), displaying Bluetooth device names [Fatah gen Schieck et al. 2010], and various information about the place itself [Churchill et al. 2006].

The above paragraph illustrates the diversity of existing systems. However, the analysis of previous work allows us to extract an initial set of design guidelines[1]. The importance of *co-realizing the system with the community* for whom the system is being built has been stressed throughout prior research [Izadi et al. 2005; Jones et al. 2008; Redhead and Brereton 2009; Taylor and Cheverst 2009; Rubegni et al. 2011] and implies that without gaining insight and understanding into a particular community's needs, the system is unlikely to be supported by the community. The importance of *building systems on top of existing behavior and practice* has also been stressed [O'Hara et al. 2003; Alt et al. 2011a] in order to avoid community members being required to change existing habitual patterns of action. It has also been recommended that the display's *purpose is made clear* [Clinch et al. 2011; Munson et al. 2011] and that the *location* of the display is kept in mind, as this can strongly influence how its function is perceived [Snowdown and Grasso 2002]. For example, if the display is located near a workplace it is more likely to be associated with work, whereas if it is located near a cafeteria it is perceived to convey more leisure content. The distance between viewers and a display also plays an important role, as larger, far-away displays typically do not invite people to interact directly with the display.

Typically, from the outset, public displays will need to come already filled with content [Storz et al. 2006; Taylor and Cheverst 2009]. Furthermore, the availability of a number of strongly motivated initial users will help spark community interest in the system's use [Izadi et al. 2005]. Taylor and Cheverst note that promoting such a system with a community event explicitly organized around it can also help to jump–start its acceptance. Storz et al. stress that creating original content is difficult and that having a ready source of existing content simplifies adoption as people are typically more comfortable when they interact with or manage already existing types of content [Storz et al. 2006]. An obvious solution to this problem is to solicit user-generated content [Lévesque et al. 2006; Taylor and Cheverst 2009; Alt et al. 2011;

Fig. 1. a) The rural village of Wray (left), b) Interaction with early Village Hall deployment (center), c) Photo Display in the Village Post Office (right).

[1] Our previous work on the Interacting Places Framework [Memarovic et al. 2012a] attempts to conceptualize the design space and provide a holistic overview of the challenges that designers and developers of such systems face.

Schroeter et al. 2012] but the authors have also investigated auto-generated content as an alternative [Memarovic et al. 2011b]. Schroeter et al. examine the “sweet spot” between people, content and display location and describe how these three parameters influence the users’ quantity and quality of feedback to a civic-related topic [Schroeter et al. 2012]. Similarly, the Memarovic et al. have also explored content preferences for networked public displays for a student community and how networked public displays would ‘fit in’ within the existing information and communication technologies, namely, social networking sites, email, and instant messaging services [Memarovic et al. 2012c].

The layered framework presented in this article complements the aforementioned ad-hoc design guidelines by structuring and grouping challenges around the development of public display systems which have stimulating community interaction as a primary goal.

1.3 Reported Deployments and Their Settings

Apart from analyzing the reported literature, we mainly draw from our own experiences with deploying public display applications for community interaction – one in a rural context and two in an urban context.

The Wray Photo Display (described in detail in section 2) was deployed in a rural village in the North of England (see Fig. 1-a). In August 2006, the first technology probe based photo display was deployed in the Wray Village Hall (see Fig. 1-b). Subsequently, and due to refurbishment of the village hall, the photo display was moved to the village shop (Fig. 1-c), where it is still in use (January 2013). An additional display was deployed in Wray’s only café in February 2010. In 2011 this display was moved to the village pub, where it is still in use.

FunSquare (described in section 3) and Digifieds (described in section 4) were deployed on the UBI-Hotspots public display infrastructure [Ojala et al. 2010] as part of the International UBI Challenge competition [Ojala and Kostakos 2011]. The UBI-Hotspots are twelve networked public displays equipped with two overhead cameras, an NFC/RFID-reader, a loudspeaker, WiFi and Bluetooth access points, and high-speed Internet access (cf. Fig. 2). They were installed throughout the city of Oulu, Finland, in June 2009 and have been in continuous operation since.

Fig. 2. a) Outdoor UBI-Hotspot in the pedestrian area in Oulu, Finland (top left), b) Indoor UBI-Hotspot in the public library (bottom left), c) UBI-Hotspot user interface in the interactive mode (right, from [Ojala et al. 2012]).

2 EXPERIENCES FROM THE WRAY PHOTO DISPLAY DEPLOYMENT

The Wray Photo Display is a public display system that allows the local community to upload, moderate, and categorize photos that are shown on a public display. It was co-designed with significant engagement with the local community using a technology probe based approach [Taylor and Cheverst 2010].

The Wray Photo Display was initially deployed in 2006. The goal of the deployment was to investigate the effect of such a public display system on the sense of community in a rural community setting. Our hypothesis that sense of community would be positively affected came from user feedback received during previous photo display deployments (e.g., [Cheverst et al. 2005]) and from the theoretical understanding that a key element of sense of community is shared emotional connection (e.g. shared sense of history) as posited by [McMillan and Chavis 1986].

The system's evaluation involved qualitative and quantitative aspects. Qualitative evaluation/feedback came through focus groups, design workshops and a comments book placed next to the display(s). Quantitative aspects included interaction logs and basic content analysis, e.g. to date over 2200 photos have been submitted across 36 content categories.

The system has undergone a number of revisions, some on the level of new functionality (based on user feedback) but others at the level of hardware or system architecture (cf. sections 2.1 and 2.2 respectively). It is important to note that for this deployment we – the researchers – had full control over these factors, i.e., there was no requirement to use a particular, hardware, system architecture, etc. The system is still up and running in the village of Wray.

2.1 Hardware

To allow a reduced development cycle and rapid deployment as a probe, the Wray Photo Display was constructed from off-the-shelf components. A Mac Mini was chosen to run the system, due to its near-silent operation, small form factor, and relatively high specifications. In some of our earlier display work significant time resources were spent on assembling a near-silent small form factor PC [Cheverst et al. 2003] and so the availability of the Mac Mini was welcomed. A simple, single-point, resistive touch screen display was used to display the content. Using these off-the-shelf components meant that we had stability/reliability with the display hardware from the first deployment. However, in terms of Internet connectivity, this was not the case: the Wray display utilized the village's wireless mesh network, which had been installed as part of a previous research project. Networking over household power lines was used to connect rooftop wireless receivers to the display itself. However, this was an experimental network where outages and low bandwidth were frequent. As a result it was important not to assume always-on Internet connection and consequently we had to consider how to show display content offline (discussed in the next section). The stability of the network improved significantly during the course of the deployment.

2.2 System Architecture

Similarly to the display's hardware, the system architecture was designed to provide reliability and stability, while also allowing rapid iterations and incremental improvements. It is important to note that for this deployment, we anticipated using only a small number of displays and therefore considering issues such as scalability (i.e., the ability to support a large number of networked displays) through the system

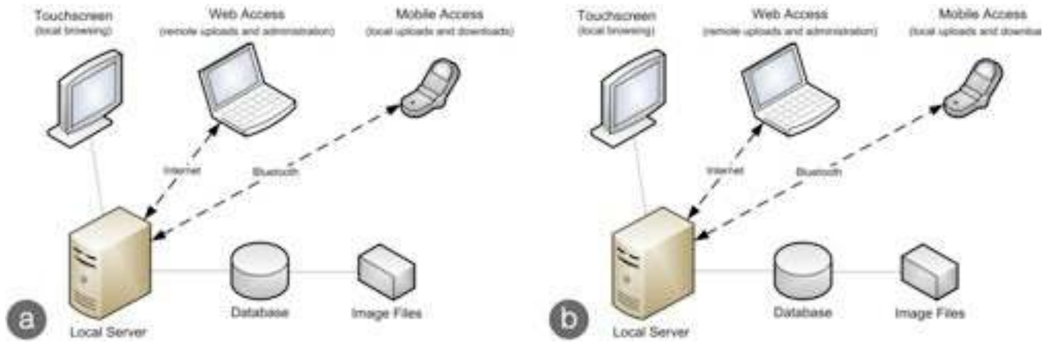

Fig. 3. The initial (a) and revised (b) system architecture of the Wray Photo Display.

architecture was not a focus (and this contrasts with the scalability support provided by the UBI-Hotspots based systems discussed in sections 3 and 4).

The software was developed as a full screen Java application, which displayed a locally-hosted collection of photographs and allowed users to navigate through the collection. To support our iterative development approach, the software architecture was designed to separate the user interface from the underlying data model, allowing rapid redesigns of the user interface without the need to re-engineer the entire system.

As mentioned in section 2.1 above, at the start of the project network connectivity was intermittent. Consequently, in order to ensure that the display would be able to show content (during a period of network unavailability) the initial system architecture (as shown in Fig. 3-a) was designed such that all content was hosted locally on the display machine itself. Villagers could log directly into this machine through a web interface to upload and browse photos, but only when the network was available.

Two years into the project, the architecture was significantly revised (see Fig. 3-b) such that all content was hosted on a central server, and the Wray display would act as a client that cached a local copy of the data and synchronized at regular intervals. This approach opened up the possibility of allowing more than one display to be deployed, and for each display to present the same content, while retaining external access to the photos should the displays / network fail.

2.3 Content

The Photo Display supported a 'photo gallery' style application (see Fig. 4), which was initially seeded with photos of the Wray scarecrow festival taken by one of the authors. Subsequent content was provided directly by community members themselves, who uploaded the content through a website. This website also provided a mirror of the display's content, allowing it to be browsed from home.

The early technology probe displayed photos in a flat hierarchy, i.e., without any classification. However, feedback from the community suggested several content categories, e.g., historical photos and funny photos and videos. While supporting such additional content categories appeared an obvious choice, the harder decision was how to support moderation of the growing array of content. This was especially important given the public nature of the deployments, but also because we wanted to foster a sense of content ownership by the community. After consultation with community members, a joint decision was made to enable individual community members to request and 'own' categories. In other words, each member that created a category would have responsibility for moderating the content in that category. At first we were worried that this type of moderation could create problems within the community. For example, certain residents could have had personal differences with the project champion – the research team's contact person in the community – or the person in charge of a given category. However, over the years the authors have not been made aware of any such problems.

While support for content categories was the first revision of the system relating to content, others included enabling comments to be posted on individual photos using an onscreen keyboard and allowing photos to be sent as 'e-cards'.

2.4 System Interaction

System interaction was informed by the placement of the displays. For example, village residents or visitors to the village could notice the display and its content while waiting for their doctor's appointment in the village hall (which acted as a doctor's surgery on set days a week), browsing the local shop, or queuing to place their order in the café. This influenced our decision to support passive interaction at first followed by a more engaging one (if the users decide so). Users could passively observe display content that scrolled through photo galleries at 20 seconds intervals. On noticing the content they could interact directly by browsing through categories and selecting pictures and their associated descriptions.

Many residents in Wray were typically comfortable with only a small number of technologies that they used regularly, and were often hesitant around unfamiliar ones. For this reason, and in an attempt to lower the barriers to use and encourage interaction by a large number of residents, the display's interface was designed to be as simple as possible. Functionality was kept to the minimum required to fulfill the community's needs, and elements were removed if they proved redundant or unpopular, as was the case with a feature that allowed uploads by Bluetooth.

2.5 Community Interaction Design

Fundamental to the Wray Photo Display community interaction design was the freedom to allow village residents to post whatever content they find appropriate and thus share their values and interests. This was also reflected in the (joint) decision to allow community members to define and 'own' categories. The following example best illustrates the type of community interaction Wray Photo Display supported. A simple comment requesting more historical content on the display led to a large number of historical photos being scanned and uploaded. These photos were uploaded in two new categories "Old Photos" and "Wray Flood" showing the village's history and fo-

Fig. 4. Gallery screen displaying images from the Scarecrow Festival 2008.

cusing on a flash flood that caused significant damage in the 1960s respectively. Many of these photos had previously been held in personal collections and had not been made public. Because of the Wray Photo Display these photos were circulating in the public, educating younger villagers of local history and reminding older ones of times when joint community spirit helped overcome natural disaster.

The flexible nature of the system that allowed community members to choose what type of content they would see was further extended with the photo commenting option. This option acted as a 'blank canvas' and gave the community freedom to choose the role that it played. Emergent uses included identifying people and events shown in the historical photos, appealing for information about photos, making jokes, and simply expressing opinions about photos and events that were shown.

3 EXPERIENCES FROM THE FUNSQUARE DEPLOYMENT

FunSquare [Memarovic et al. 2011b, Memarovic et al. 2012b] is a service that explores the potential of public displays to stimulate the effect of "social triangulation" in public spaces. Social triangulation is an effect where unusual features in the space, such as sculptures, fountains, or street performance, stimulate people in the vicinity of the feature to engage into spontaneous social interaction [Whyte 1980]. In turn this creates the sense of social connectedness.

In order to create an unusual feature in the space, FunSquare uses "autopoiesic content" – automatically generated situated content [Langheinrich et al. 2011]. Autopoiesic content is created by matching a piece of information that is coming from within the display's vicinity, e.g., the current wind speed in the city (12 m/s), with information coming from without, e.g., the speed of a honey bee (6 m/s), and merge the two into a novel piece of information, e.g., "The current wind speed in the city (12 m/s) is twice the speed of a honey bee (6 m/s)". This information – a fun fact – represents an intellectual challenge in the space, which in turn should provoke people to engage in spontaneous social interaction [Memarovic et al. 2012b].

FunSquare runs in two modes: an *ambient mode*, where fun facts are continuously shown to provide a backdrop for conversation (cf. Fig. 6-a), and a *game mode,* where fun facts are displayed as a trivia quiz to encourage collaborative interaction (cf. Fig. 6-b). During the UBI Challenge competition, we evaluated FunSquare in both modes. Ambient mode evaluations were done by having team members observe passers-by (taking notes and photos) and conduct open-ended walk-up interviews. Evaluation of FunSquare in game mode consisted of both a quantitative and a qualitative evaluation: we logged all screen interactions within the game on the central server, and also performed user trials in the wild by encouraging passers-by to play and distributing questionnaires to them afterwards.

The FunSquare system development targeted the prescribed hardware, system architecture, and system interaction as defined by the UBI-Hotspots. In other words, for the FunSquare system the research team did not have full control over these factors, whereas with the Wray Photo Display system the researchers did. Having originally been designed only as an ambient display application, the technical inability of the UBI-Hotspots' system architecture to support the envisioned deployment (more details in section 3.4) prompted us to introduce the game mode. The new mode prompted ad-hoc changes at the system architecture, system interaction, and community interaction design levels.

3.1 Hardware

The development of the FunSquare application started in 2011, i.e., two years after the UBI-Hotspots were distributed in the city. The central component of the UBI-Hotspots is a 57-inch high-definition LCD screen with a capacitive touchscreen foil. All applications were running in a Mozilla Firefox 3.6 web browser (see also [Ojala et al. 2010] for details). For content distribution, FunSquare relied on the publicly available free Wifi network (called *panOulu* [Ojala et al. 2010]).

Most of the application development was done in a lab provided by the UBI Challenge organizers. The biggest difference between the "in lab" and "in the wild" hardware was the technology used to detect screen touches. While the lab touch screens used infrared to detect touch, the "in the wild" hotspots used capacitive touch foils. Also, the lab screens featured a dedicated Internet connection while "in the wild" devices had to rely on the panOulu WiFi. These small differences caused problems during the deployment phase where the users commented that the display was "inaccurate, hard to use" or that the application was "nice, but reacted a bit slowly" (cf. section 5.1).

3.2 System Architecture

As FunSquare was developed for the UBI-Challenge competition, it was clear that the system would have to operate across a wide city network of public displays, spanning more than a single location. At the same time, each location required its own localized context information in order to create a "fun fact" with local relevance.

The FunSquare system architecture consists of four components (c.f. Fig. 5): 1) a *context sensing component* which collects dynamic information about a display's surrounding environment and turns them into context streams, 2) a *content fragments database* that contains a large number of manually collected (fixed) facts, 3) an *autopoiesic matching engine* that combines content fragments and context streams into new content – *fun facts*, and 4) a *user interface* visualizing the fun facts. All the components were developed using standard Web technologies such as HTML, CSS, PHP, JavaScript, Java, and MySQL (see also [Memarovic et al. 2011b]). The context-sensing component interacted with UBI-Hotspots hardware sensors over a custom-made RESTful API that changed during the development. This meant that in order to keep on having access to the sensors the researchers had to follow the required updates and change the code every time there was a new version of the API (cf. section 5.2).

Merging each per-display context stream with the centralized content fragments database, the central matching engine continuously creates customized fun facts for each individual display. Every fun fact has an implicit rank, based on its age (older facts get "stale"), the type of context it uses (e.g., temperature data "ages" better than wind speed data), the "fit" of the two matched content pieces (e.g., small multiples are preferable to large multiples), the number of previous uses of the fragment and its category (in order to prevent overuse), and available user feedback in the form of likes and dislikes of the particular content fragment category. For each location FunSquare simply displays the highest ranking fun fact (more details about the selection procedure can be found in [Memarovic et al. 2011b]). Finding the best balance between the factors consumed significant development time and required multiple trials (cf. section 5.2).

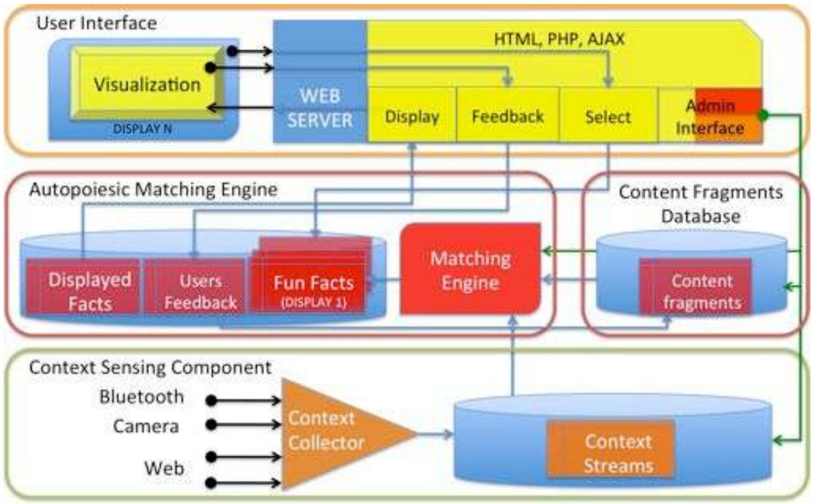

Fig. 5. FunSquare system architecture.

3.3 Content

FunSquare uses two simple templates to show assembled fun facts, depending on its mode. While in the ambient mode the system only displays the facts (Fig. 6-a), the game mode shows them in the form of a trivia quiz where users have to connect two matching pieces of information (Fig. 6-b). The "ambient" template consists of five elements, as illustrated in Fig. 6-a: 1) images that represent the content categories, 2) the matched content snippets, including the measured/matched values, and 3) the relationship between the two items. A similar template is used for FunSquare in game mode, where the template is divided into four parts: one for the question and three for the possible answers (cf. Fig. 6-b).

3.4 System Interaction

FunSquare's system interaction had to be built on top of the existing UBI-Hotspots system interaction. The UBI-Hotspots have two interaction modes: 1) passive broadcast and 2) interactive mode. In the passive broadcast mode the large LCD display shows full-screen advertisements (the so-called "UBI Channel"). If the overhead cam-

Fig. 6. FunSquare user interface for a) ambient and b) game mode.

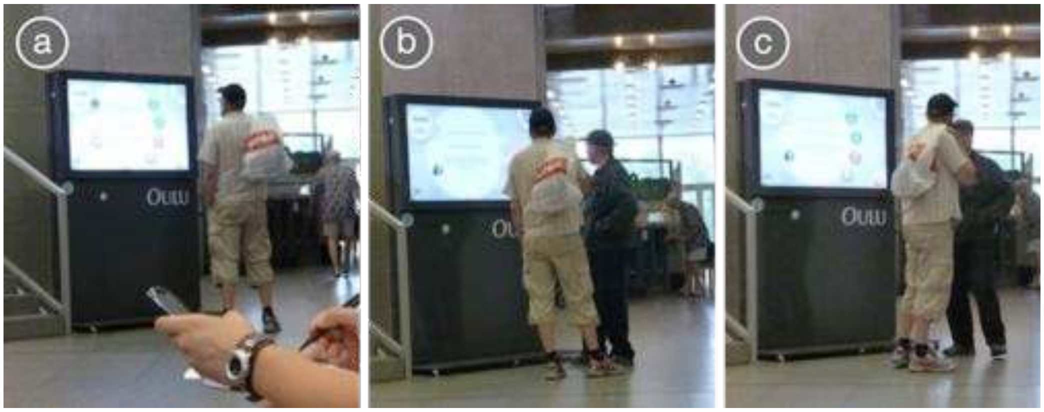

Fig. 7. An example of triangulation effect between strangers: a) a person reading fun fact, b) another person approaching, and c) the two persons starting social interaction.

eras detect a person standing in front of a display, an unfolding “touch me” icon appears in the upper right corner of the screen. If a touch is detected anywhere on the screen, the UBI-Hotspot switches from passive to interactive mode. In interactive mode the screen is divided into four parts, as shown in Fig. 2-c: 1) the UBI Channel is moved to the upper left corner and continues to show advertisements; 2) a quick-launch menu shows shortcuts for several featured applications (lower left corner); 3) a comprehensive service menu lists all available applications, grouped into seven categories (right part of the screen); and 4) a footer menu shows a clock and allows for service voting, returning to the “home” screen, logging into UBI-Hotspots, and selecting the language. We had no control over the UBI Channel and the footer menu. When an application is selected, it appears in the right half of the screen, replacing the service selection menu.

Originally FunSquare was envisioned to run in a screen saver fashion (“ambient mode”) where fun facts are continuously shown to provide a backdrop for conversation. For this purpose the UBI-Hotspot’s passive broadcast mode seemed to be an obvious choice: people could view the information when passing by a display and decide whether or not they want to interact with the sparse user interface shown in Fig. 6-a, e.g., to advance to the next fun fact or to give feedback. Unfortunately, the dynamic injection of interactive content was not supported within the UBI-Hotspot’s passive broadcast mode. Running FunSquare as a full-screen app in interactive mode was equally problematic, as it required passers-by to first leave the screen-saver of the passive broadcast mode and then select another screen-saver – FunSquare – from the menu. As this became only apparent a few weeks before the schedule deployment phase, we had to quickly create FunSquare’s new game mode in order to salvage the work. In the game mode, fun facts are displayed in the form of a trivia question as can be seen in Fig. 6-b. Players have to give as many correct answers as they can. As depicted in Fig. 6-b each question has three alternatives where only one is correct.

3.5 Community Interaction Design

FunSquare’s ambient mode aims to increase community interaction by stimulating the effect of “social triangulation” – an effect where unusual features in the space trigger social interaction. An example of how social triangulation can be triggered by display is shown in Fig. 7.

Fig. 8. Screenshots of the a) neighborhood selection and b) score presentation

In FunSquare's game mode, players had to select a neighborhood before a game (Fig. 8-a). This was meant to stimulate a sense of belonging to a community. After each game FunSquare not only displays the current player's score but also the aggregated neighborhood score and how much they contributed to it (Fig. 8-b).

4 EXPERIENCES FROM THE DIGIFIED DEPLOYMENT

Digifieds [Alt et al. 2011b, Alt et al. 2013] is a digital public notice board that allows classified ads to be created, posted, and retrieved from public displays. Besides directly creating and retrieving content at the display, Digifieds provides a mobile application that supports different ways of exchanging content with the display – phone-display touch, QR codes, and alphanumeric codes. Inspiration for the Digifieds system came from our ethnographic pre-study on practices surrounding more traditional/analog public displays such as shop windows, wall hangers and notice boards [Alt et al., 2011a]. The pre-study showed the importance of traditional notice boards for community interaction by supporting exchange within local community.

Similar to FunSquare, Digifieds was deployed on the UBI-Hotspots infrastructure. Between July and August 2011, we performed a variety of evaluations, including observations, interviews, and a field trial. The observation and interviews were conducted on the 11th and 14th of July 2011. The field trial ran over two weeks from 1st until 12th of August 2011.

4.1 Hardware

Digifieds was deployed on the same hardware as FunSquare as described in sections 1.3 and 3.1. The original version of the Digifieds application includes the above-mentioned phone-display touch feature. This feature allows content to be exchanged with the display by hitting the desired location on the screen. Unfortunately, plans to explore this technique in the Oulu deployment had to be dropped because the display hardware in Oulu used capacitive touch screens, which made it impossible to detect touches from a mobile phone. Furthermore, we discovered that Nokia is still quite popular in Finland, which excluded users that did not own an Android smart phone.

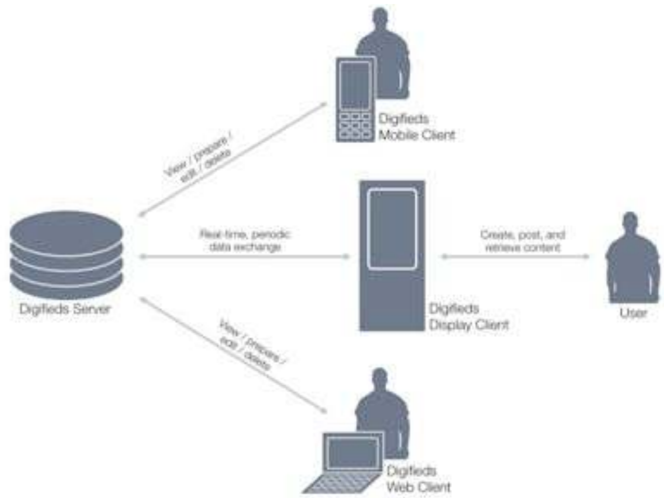

Fig. 9. Digifieds conceptual system architecture: The Digifieds server stores and maintains the content. Mobile clients, display clients, and web clients can connect to the server in order to exchange content.

This led to less use of the mobile client than we had expected, as people mostly owned phones running Symbian, and more recently Windows Mobile.

4.2 System Architecture

The Digified's system architecture consists of a client-server infrastructure that allows an arbitrary number of display and mobile clients to be connected (see Fig. 9). The Digifieds server as the central component of the system is responsible for data management and storage. As data and configurations are stored centrally, the exchange of content between clients as well as the replication (e.g., same content on multiple displays in the same neighborhood) is easily feasible. Access to the different clients is provided through a RESTful API.

The core component of the system is the *Digifieds server*, which stores and maintains the content. It requires the open source Glassfish application server, sufficient storage, and a permanent connection to the Internet. In addition, different devices can be used as clients, including displays, phones, or laptops.

The *Digifieds display client* (see Fig. 10) – a web application that renders the content and provides means to add and retrieve arbitrary classifieds – can be accessed through any browser simply by calling a specific URL. To allow classified ads to be directly created at the display, a touch-enabled surface is required so that the users can enter information via an onscreen keyboard. As an alternative, content can be accessed also from any personal computer through the Digifieds web site where the alphanumeric code (cf. section 4.4) associated with each classified ad can be entered.

To support easy content transfer to and from a public display as well as to create content on-the-go, the system provides the *Digifieds mobile client* – an Android app that is available from Google Play or a URL shown on the display. An Android phone is required for installing the client. Optionally, the camera of the mobile phone can be used to embed photos and videos into the post as well as to exchange content with the display via QR codes (retrieving content is either using a native QR code reader or the QR code reader integrated with the application).

Fig. 10 – Digifieds user interface: The display client shows an overview of the different views on the left, the active view, containing the actual digifieds, on the right.

4.3 Content

Digifieds has been designed to support all types of content that are commonly being found on public notice areas, including, but not limited to, job offers, announcements of events, sales offers and housing. Whereas posts on digital public notice areas are usually not categorized but rather randomly distributed across the display (offers for babysitting are found next to event flyers and items that are for sale [Alt et al. 2011]) we enable each classified to be associated with a certain category in order to enhance search and retrieval. The moderator of the display or the group of displays (cf. section 4.4) configures the categories supported by a display in the Digifieds network. The digified itself can, in its most basic form, consist of text only, but advertisers can enhance their posts with photos or even videos.

During the 2-month evaluation period, 49 digifieds were posted in the eight provided categories. “Sales” was the most popular (23 items), followed by “Jobs” (10), and “Events” (8). These posting preferences are similar to those revealed from our previous study on traditional PNAs [Alt et al. 2011a]. Housing was not as popular on Digifieds (3% of all items) as it was on the traditional PNAs (15% of all items), which might have been a result of the fact that the evaluation was conducted in the main holiday period. As Oulu is a student city, interest for housing may grow as soon as the new term starts. Similar to content posting preferences, “Sales” and “Events” were the most viewed categories. Summarizing the findings, it can be seen that the envisioned content as well as the actually posted content correlates both for traditional as well as for digital PNAs.

4.4 System Interaction

Whereas creating content directly at the display only requires simple touch interaction using the on-screen keyboard, we needed to employ more complex interaction techniques for exchanging content between the display client and the mobile client (see Fig. 11). Users can exchange digifieds between display and phone by scanning the QR code of the associated digified (see Fig. 11-b) by using an alphanumeric code (see Fig. 11-c) or through the phone-display touch technique (cf. section 4.1). Note that the latter one, as previously mentioned, was not supported by the hardware during the field deployment. In addition, we enable content to be retrieved via email

Fig. 11. Exchanging Content: Users can send digifieds to their a) email address or transfer them to their mobile phone using either the b) QR code, or the c) alphanumerical code.

with the address entered via a form (see Fig. 11-a). The user is then sent the content and the URL of the digified.

For retrieving multiple digifieds, we employed a "shopping basket" metaphor for the display client. By using this, users could collect all interesting classifieds before transferring the entire collection to their phone or sending it to their email address. We also enabled a search functionality to look for keywords. For simplicity, the search functionality masks out all digifieds that do not contain the required keyword.

4.5 Community Interaction Design

The ethnographic study of traditional PNAs revealed that locality played a major role. Most of the content was targeted to local communities (e.g., advertisements for a concert in a certain area of town, course books for other students, etc.). In addition, we found many examples, where this information was distributed manually over multiple displays that were co-located (e.g., in the same neighborhood / building / etc.). In order to preserve this local character, we designed a concept called *display groups*. Each group comprises a number of displays with certain properties, as depicted in Fig. 12. In this example, displays are grouped based on location, e.g., all displays at the market square (Area A), all displays in the pedestrian area (Area B), and all displays at the sports center (Area C). However, this concept is not limited to location. The owners of a display can define their own groups and assign the displays. In this way, displays can be grouped based on the needs of the community (e.g., all displays that are in the vicinity of skater parks, etc.).

Fig. 12. Display Groups: To preserve the locality of content we created a concept called display groups. Displays are grouped based on location, e.g., market place (blue), pedestrian area (red), and sports center (yellow).

5 THE P-LAYERS FRAMEWORK

The three summaries in sections 2-4 of our own public display deployment efforts hopefully illustrate the obdurate problems associated with stimulating, capturing, and examining community interaction effects "in the wild". This difficulty is perhaps best captured in a quote from a FunSquare game user: "OK idea, bad execution." In retrospect, we can identify five main challenges researchers need to address in these deployments. In many cases, the *hardware* hindered the smooth operation of the system. We also underestimated the complexity of the *system architecture*. Getting appropriate and fresh *content* that is appealing for the community was challenging, and offering intuitive ways of *interacting* with the system – in particular for passers-by – continues to be a problem. All of these factors affected what we were primarily interested in evaluating: actual *community interaction*. We can layer these five factors into a framework that describes challenges of building public display systems that support community interaction: the P-LAYERS framework (from "**P**ublic display **LAYERS**", pronounced 'players'), as shown in Fig. 13.

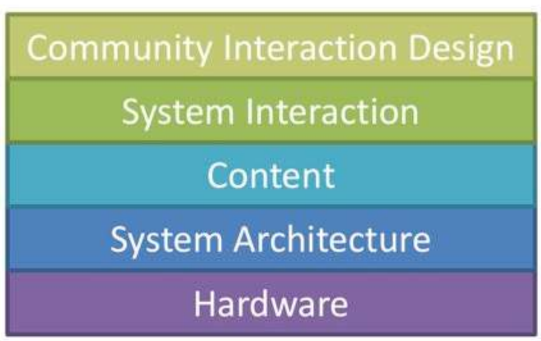

Fig. 13 – P-LAYERS framework for addressing the multi-faceted issues facing community-supporting public display deployments.

The framework attempts to capture the difficulties intrinsic to building and assessing public display systems that aim to foster community interaction "in the wild". In the following sections, we provide a detailed explanation of the different layers of the P-LAYERS framework, starting from the bottom. For each layer we will present a joint summary from our development and deployment experiences.

5.1 Hardware

Hardware sits at the bottom of the framework, signifying its fundamental importance as a foundation for any display based deployment. If the hardware does not fulfill requirements and expectations, both from users and the researchers, higher levels will be affected. In our deployments, three main insights emerged:

1. The importance of matching development and deployment hardware,
2. The importance of communicating screen affordance, and,
3. The reliability of hardware components and availability of replacement parts.

Having the same development and deployment hardware is critical since any differences between the two can lead to contrasting user experiences. For example, the Wray Photo Display had exactly the same hardware for development and deployment. In contrast, for FunSquare and Digifieds, the hardware used for development in the lab was different from that used during deployment. These differences between lab and "in the wild" setup resulted in very different user experiences in the two settings. In the case of FunSquare, once the application was developed and moved from the lab setting to the UBI-Hotspots installation, one of the most frequent complaints from users was that the touch screens were "inaccurate, hard to use" or that the application was "nice, but reacted a bit slowly". These problems were hard to spot during our test trials in the lab, since the lab had a later version of the hardware and a more reliable Internet connection (which was required by the FunSquare application). Similarly, the novel phone-display touch interaction modality developed for Digifieds that supported the transfer of content through touching the screen with the phone (cf. section 4.1) had to be dropped due to the use of capacitive screens "in the wild" as opposed to the lab, where resistive touch screen were available. The reason was that only the resistive touch screens available in the lab were able to detect touches from a mobile phone. This immediately eliminated the potential use of this interaction modality in Oulu.

Once the system is rolled out "in the wild", proper performance depends on the reliability of the hardware components. For example, in all three of our deployments there were considerable issues with Internet connectivity that impacted on user experience. Both FunSquare and Digifieds were using the publicly available *panOulu* free Wifi network. Occasionally, bandwidth decreased or connectivity broke during peak hours, i.e., when the citizens used the network most – these peak hours usually overlapped with those of the UBI-Hotspots. Since fresh content was fetched over the Internet, lower network throughput created "jittery" interaction with the system, which lead to a frustrating user experience. Similar problems were encountered in the early phases of deployment of the Wray Photo Display, where an experimental mesh network was used in the village and the early system architecture required good levels of connectivity.

The central hardware component in the system is the display itself. Therefore it is important to consider how to communicate its affordance to users. For example, the resistive touch input featured on the UBI-Hotspots in Oulu were very much in contrast to what can be found on today's smart phones and other personal devices that have high-quality capacitive touch screens. Most users expected to get the same user experience as they had with their mobile phones and were not satisfied if the screen did not provide the same experience. User expectations might have been better aligned with the displays' capabilities if the design was such that users were aware that the touch screens were not as sensitive as the ones they are used to [Chalmers and MacColl 2003].

Even reliable hardware has a certain lifespan. For that reason it is important to plan for replacement parts. This is especially true for long-term deployments that include full transfer of system operations to the community. For example, in case of the Wray Photo Display, a hard drive failure occurred for one of the Mac Minis. This caused issues for sustainability and handover to the community [Taylor et al. 2013]. Hence it is advisable to check hardware reliability and/or the warranty period and, for contingency, ensure that compatible hardware is still available if replacements are required.

5.2 System Architecture

The overall system architecture of a public display system for supporting community interaction can appear straightforward: a touch screen as an I/O device, a local computer running a Web server or similar digital signage software, and an Internet connection for remote administration. However, by going beyond traditional digital signage systems that only need to play pre-determined content, two major new challenges for the system architecture arise: *interactivity* and *durability*. Interactivity not only means the direct user-to-screen interaction, but also interaction between different deployment sites, or multiple interaction capabilities (e.g., via touch screen, phone, and Web). Durability refers to the fact that – ideally – such deployments will run for months, if not years, and thus need to take into account long-term maintainability. Four main issues that impact interactivity and longevity emerged from our deployments:

1. System scalability.
2. Agility to follow changes in third party services/browser.
3. The challenge of finding the right level of complexity.
4. The challenge of supporting appropriate interaction modalities.

The issue of system scalability is best contrasted with the two deployment settings, i.e., Wray and Oulu. From the beginning the system architecture of FunSquare and Digifieds had to be adapted for a citywide display network, which could potentially represent hundreds of displays. In contrast to this, the Wray Photo Display deployment did not need to consider scalability: the focus of the Wray Photo Display was more on the technology probe based deployment within an iterative user-centered design process. The Wray Photo Display's system architecture was originally designed for a single display. However, once more than a single display was needed the system architecture had to be modified for the new conditions. In other words, the architecture had to support decision making for *where*, i.e., on which display to show content. Still, system scalability can go beyond just decision making of where to show the content. In the beginning all the pictures uploaded to the Wray Photo Displays were stored on the local computer. It is not hard to imagine that if the success of the system was sudden this would not scale: if there were hundreds or thousands of users uploading their pictures the local computer would quickly run out of storage. Considering such a situation from the beginning would have required a different approach where pictures would be uploaded to a cloud based service or a professionally managed one.

However relying on third party services carries its own issues. In order to access available sensors from the UBI-Hotspots, FunSquare relied on custom made RESTful APIs, one per sensor. During FunSquare development the APIs were further developed and updated. This meant that whenever there was a change in the parameters received from the respective service this had to be reflected in the code in order to ensure that content coming form the service would be received. Also, UBI-Hotspots were running on a specific browser version of the Mozilla Firefox browser (3.6). This also had to be reflected in the code. If the browser version on the hotspots was updated this change would also have to be reflected in the code as well. Upgrading to the latest browser version on the UBI-Hotspots would allow the use of the latest web technologies, e.g., HTML5. However, considering that the system architecture of the UBI-Hotspots was built when the specific version was the latest one (and that all the applications running there are built for it) upgrading to the latest version would cause major problems for the system. These examples highlight the need for agility to follow changes in third party services and software, e.g. browser versions.

The above examples also illustrate some of the choices that can influence the complexity of the system architecture. An obvious rule of thumb for finding the right level of complexity would be to *start simple and add complexity later*. This was most evident with our FunSquare ambient mode deployment. During development, we spent a considerable amount of time brainstorming on how to display the most appropriate "fun fact" for a given situation. The ranking system we came up with (for more details see Memarovic et al. [Memarovic et al. 2011b]) ended up using a large number of factors (unit, numerical magnitude, timeliness of the context information, overall usage of a content category, number of uses of a particular content fragment, and user feedback). This added to the complexity of the overall architecture, both in terms of the decision process (algorithm) as well as for data management (meta data). In our subsequent lab tests, the selection procedure seemed to work well. However, during observations and interviews, it turned out that most people had clear preferences towards certain categories and would have liked a simple category-selection mechanism. While our complex selection process worked, a much simpler manual system might have worked just as well, with much lower complexity and more sustainable durability.

Finding suitable input and output modalities for interaction can lead to longer durability. For public display systems this would include finding the appropriate way(s) of *how* (in what form) and *where* (on what device/display) to present content. In Digifieds, *how* information was presented depended on *where* it was accessed. For example, when digified was presented on a display client it included high-resolution images, while on the mobile phone downsized images were used. Similarly, the display would offer various controls for retrieving content, as well as a 'like' and an 'abuse' button – none of which was needed for the mobile client.

5.3 Content

"It's the *content*, stupid!" one is tempted to state, slightly adapting a well-known U.S. election campaign phrase. As Clinch et al. point out, content creation is one of the most underestimated resource costs in digital signage systems [Clinch et al. 2011]. Given the envisioned long-term deployments and the strong need for content that resonates with the target community, four challenges arise:

1. Finding and accessing appropriate sources for content.

2. Determining a suitable content format.
3. Identifying the meta-data requirements for the content, given a particular setting.
4. Managing content, both by users and by system administrators (moderators).

Appropriately seeding content needs to be resolved before a public display systems rolls out into “the wild”. The three services that we worked on portray two different choices of seeding content. FunSquare represents a public display application that uses content from a *service* by connecting two different content items (i.e., information that is sensed within the display and information that is stored in a database). On the other hand, both Digifieds and the Wray Photo Display required people to post their own text and images, i.e., they both rely on *people/user-generated content*. Both approaches have advantages and disadvantages. Neither of the two choices is inherently better suited or easier to use. User-generated content requires an initial seed phase where the system is seeded with content, as users are less motivated to fill an empty system (cf. section 2.3). A service-based content system, on the other hand, needs to ensure that its content stays fresh and relevant, as it does not enjoy the benefit of community members themselves updating it.

Determining the suitable content format is equally important as resolving the appropriate content source. As mentioned at the end of the previous section, Digifieds had a high and a low resolution version of a given image to be used depending on where the digified was shown: high-resolution image for the display client and low resolution one for the mobile phone. Another important property of the content format for Digifieds and the Wray Photo Display is that they supported open and commonly used standards, e.g., JPEG. These ensure widespread use and audience reach. Considering the support for the latest content types is also important since it can have a big impact on the system architecture. For example, if an application requiring HTML5 content, e.g., audio or video through the getUserMedia tag, was about to be deployed in Oulu the system architecture would need to change to the latest browser version that support this.

Once the public display system for stimulating community interaction is out and running its content has to be dynamically selectable and adaptable for different situations and communities. In order to allow content to adapt we can augment it with meta-data. Meta-data can allow for: 1) better content distribution, i.e., the correct content appearing on the correct display; 2) expressing a community’s content preferences explicitly (e.g., FunSquare’s ‘thumbs up/down’, Digifieds’ ‘abuse’ button, or opinions posted as comments on the Wray Photo Display); 3) assessing community content preferences implicitly (e.g., in Digifieds meta-data about the number of times an ad was viewed or downloaded); and 4) allowing personalized content labeling (e.g., tagging content in Digifieds). Identifying the right set of meta-data has obvious implications for neighboring layers (system architecture, system interaction).

Last but not least, content must invariably be managed in one way or another. For both Digifieds and the Wray Photo Display, there was a need for *content moderation*. In Digifieds, users could report inappropriate content through the abuse button. During the initial six months of deployment, two items with unsuitable content were reported and consequently removed. This type of moderation allows community members themselves to gauge inappropriate content. In the case of the Wray Photo Display, a more centralized moderation solution was implemented. As described earlier in section 2.3, residents of the village could ‘own’ a particular content category, which entailed the responsibility of moderation. In both cases, however, a review delay will potentially block appropriate content from appearing: in Digifieds, a reported item would be taken immediately out of rotation until reviewed, while in the Wray Photo Display, all new content had to be explicitly approved.

The service-generated content used in FunSquare instead required a *dynamic content management* module that would ensure that content would not repeat itself too often. The module would also allow explicit moderation, as users could use "thumbs up" and "thumbs down" buttons to express their preferences for particular content items. As reported in section 5.2, however, much of the content management architecture that we initially devised turned out to be of only moderate use, as users ultimately preferred to manually select content categories.

5.4 System Interaction

Interactivity is key to allowing a display to become an active facilitator of community interaction. Three main questions need to be answered when it comes to system interaction:

1. *Where to place the display?* The location and exact placement significantly affects how users approach and interact with a display.
2. *Which level of complexity is appropriate?* Complex user interfaces support more powerful applications, yet can make interaction less obvious.
3. *How should interaction be triggered?* Users might not directly understand the interaction capabilities of a display, in particular when it involves subtle cues or advanced technologies such as NFC or Bluetooth.

In the case of the Wray Photo Display, there was some flexibility in choosing where displays would be located. The most desirable locations were the ones most frequented by residents and visitors, i.e., village town hall, post office, and café. Activities at the locations informed the way system interaction was designed: most of the users were seen *waiting* for their doctor's appointment in the village town hall or *queuing* to place their order in the post office. A key design decision was that interaction should be lightweight: people could simply observe the content without interacting with it and content would change every twenty seconds. After that, if they had more *time* they could approach the display and interact directly by browsing through categories, selecting pictures, or reading their description. This example illustrates how *activities at the location can inform system interaction*.

All three deployments supported lightweight interaction with content in the form of *content browsing*. In FunSquare's ambient mode, users were able to click on the "next fun fact" button, while Digifieds and the Wray Photo Display allowed users to switch between different categories as well as browsing back and forth between them. Although categorizing content provided information as to how it is presented and organized, it also added to the system interaction complexity where users had to perform several additional touch-clicks in order to get to their desired content.

Not all interaction capabilities might be immediately obvious to users. FunSquare had a timer in the lower right corner that showed the time left for a particular fun fact to be displayed. However, not all users understood what the timer meant. Similarly, some people did not realize that the display was interactive, others realized that buttons were clickable but did not know what they did. Several users stated that they would prefer if some instructions about the meaning of the buttons had been present. More homogeneous communities might allow very specific or simple metaphors to be used. Yet, for a general audience, textual descriptions or explicit help buttons might be required. As a solution we tried to use a QR code in FunSquare's ambient mode, which featured a surrounding text "Take this fun fact with you".

Apart from the QR code itself, no other explanation of how this fact could be retrieved was offered, as we assumed that users would be familiar with the codes. However, most users ended up trying to click on the code.

One thing to have in mind when placing interaction elements is that – depending on the display's size and position – there are display areas that users are blind to. For example, in FunSquare ambient mode (see Fig. 6-a), some people did not notice the timer in the lower right corner. In game mode, where the timer was located in the central lower area, it was similarly overlooked:

> "Big screen, you have to play too close. I didn't notice the time."

A similar issue occurred in early versions of the Wray Photo Display, where users did not notice navigation controls in the center of the display. These examples point to a specific issue with the design of system interactions for large public displays: the interplay between the display and its surroundings.

5.5 Community Interaction Design

At the top of our framework is the intended use of our intervention, i.e., the design of a display that supports community interaction. Even if all underlying layers are successfully addressed, plenty of challenges remain at the top in order to engage a community. The major challenges we experienced in our deployments were:

1. Communicating the value proposition of the application to the users.
2. Avoiding a negative impact on the community.
3. Considering interaction between different communities/stakeholders.
4. Designing for system sustainability.

The fact that a user can understand an application's interaction capabilities is not enough to ensure that they can also understand the *community interaction design*. An example observation from our FunSquare deployment illustrates this: a father and his daughter browsed through a number of facts and voted ("thumbs up") for almost all of them. In the subsequent interview, both stated that they understood how to interact with the application. Yet, they could not understand the *meaning of the application*. FunSquare's purpose was to serve as a conversation starter and its value was in stimulating social interaction. However this type of value is obscure and has to be wrapped in a more concrete and straightforward goal. For example, the accent could have been put more on the learning potential of the application. We tried to do this through the heading text "Did you know that…". However, having something more explicit, e.g., "Learn new facts about Oulu" might have made the value proposition clearer.

FunSquare's game mode was much easier to understand, yet its concept of "playing for a neighborhood" also had some unanticipated consequences:

> [How did you feel about your contribution to the neighborhood's score?]: "Not good because I didn't get any question right."

The above quote shows how the intended community interaction might actually have a negative effect if it is not achieved. While it is unclear whether such negative experience actually lowers people's involvement with a community, it might certainly deter frequent use of the application. One option might have been to provide some points for successfully completing the game, independent of the performance. Another user pointed out an additional unanticipated effect of the neighborhood game concept:

"Fun to see how own neighborhood is doing in comparison with the others. On the other hand, could aggravate the relation between the areas."

A similar experience regarding such inter-community processes comes from the Wray Photo Display deployment. One of the goals of the deployment was to support exchange within the community. In April 2010, the post office installed a coffee maker and started selling coffee drinks for take away. This 'new venture' was advertised through the Wray Photo Display. The advertisement not only appeared on the display in the post office but also on the second display, which was installed in the café. This caused a stir in the community and between the two places. After the advertisement had been noticed by the café's owner, its removal from the display was requested. This example shows that "in the wild" it is not enough to just consider community as a coherent entity but that attention has to be given to inter-community relationships and interests and the need to be wary of potentially divisive deployments in the community.

Finally, it is important to consider ways that will allow system sustainability and each of three systems had different approaches. For FunSquare, system sustainability was reflected with the type of content that was displayed – autopoiesic content – which was generated "on the fly". This approach ensured fresh content in the long run. The Wray Photo Display and Digifieds systems had different approaches. System sustainability for the Wray Photo display was conceived through the participatory design process where the community and its opinion played a key role for every revision of the system. This way the community also felt a sense of ownership for the system. Allowing community members to create and own picture categories further stimulated the sense of ownership. Digifieds adopted a similar approach for achieving system sustainability. As described earlier, classifieds uploaded to Digifieds could be restricted to a certain area where displays were available. However, such geographic grouping and filtering was actually supported in a very generic fashion, potentially allowing for arbitrary grouping and filtering (e.g., all displays in the vicinity of churches). This conscious design decision was made in order to support more fine-grained community information dissemination along a variety of factors. We believe that allowing for self-organization/appropriation by the community is key for an application's acceptance and system sustainability.

5.6 Interplay Between the Layers

Issues in one layer of the P-LAYERS framework often strongly influence neighboring layers as well, i.e., choices on one layer percolate up or down and thus restrict or open up choices in the following layers. In this section we provide a number of examples that illustrate how issues at individual levels can impact neighboring levels of the framework.

Starting from the Community Interaction Design Layer In the particular case of the Wray Photo Display, one of the goals at the *community interaction design* level was to support a sense of ownership in a fully inclusive manner, i.e., across the whole Wray community. This in turn placed a requirement on the *system interaction* layer, e.g., to allow all members of the community to upload pictures to the system. However, the fact that only web forms were available for supporting this task meant that many elderly residents struggled with uploading their pictures. Some elderly residents asked the technically competent champion in the village to do this on their behalf, but clearly some felt a social reluctance to do this. One potential solution to this, which we would still like to pursue, is to provide an effective solution at the *hardware* layer. This could, for example, involve tailoring the photocopier in the village post office to act as a simple scanner for inputting pictures into the system. This alternative was discussed with some enthusiasm by elderly residents at one of the design workshops. While still not ideal, this approach would likely provide an alternative with a significantly lower barrier to entry for certain users

Starting from the System Interaction Layer In the FunSquare game mode, community interaction was designed around a game. The game was limited to ninety seconds and users would receive an additional five seconds for each correct answer. This time limit was introduced to raise the competitive spirit and excitement within the game. However, for some users this had a very negative consequence:

> "Had to hurry up when answering. The alternatives were hard to understand."

This aspect of *system interaction* had a direct impact on the *community interaction*, as users felt rushed and did not feel comfortable playing:

> "Playing for a neighborhood is a pretty interesting idea. There could be more time to answer the questions."

> "[You] don't want to betray your own neighborhood, but [instead] get the best points you can. An OK idea, [but] bad execution."

These examples illustrate again the need for professional support. As none of the researchers involved in FunSquare had any experience in game design, the community interaction design did not live up to its full potential. Involving game designers prior to the deployment might have significantly altered the community interaction experience.

Starting from the Content Layer. Content can strongly influence people's opinion on how they can interact with it. One interesting observation in our Digifieds deployment was that people thought they would have to sign up for the service in order to be able to use it. We believe that the reason for this is the similarity of Digifieds to Web-based services such as Ebay or Craigslist – which require an authentication. This shows how *content* – particularly its design – can have a direct impact on *system interaction*, i.e., on people's perception and expectation.

While the Wray Photo Display was a novel system for the respective community, both FunSquare and Digifieds were running on previously deployed hardware where users were familiar with (existing) display content. In one particular case, two occasional UBI-Hotspot users refrained from interacting with the FunSquare application because *content* was different from that they were used to, i.e., issue at the content layer propagated to the *community interaction* layer as well. This could have been potentially avoided by paying attention to the specific user group, i.e., users who have prior experience with UBI-Hotspots. For example, this could have been done through an on-display element that would state something like "Novel UBI-Hotspots service, try it out!".

Starting from the System Architecture Layer. In the first design workshop for the Wray Photo Display, there was a request for the ability to have appropriate awareness of what content was appearing on the display at any given time – without having to be physically present at the display. The agreed solution was to have a web page that would show a screen grab of the photo display. While such a solution is trivial it created an issue within the chosen *system architecture*. It meant that the web server had to reside on the photo display itself – rather than a server at the uni-

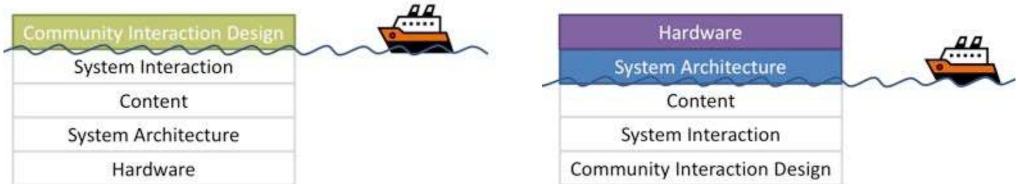

Fig. 14. Two examples of the use of the _P-LAYERS framework for self-assessment. Depending on one's individual interests and level of awareness, some layers may be deemed more challenging than others. P-LAYERS might help alerting one to unsuspected issues early in the design and deployment process.

versity – in order to ensure that the photo content would still be visible on the display even in the event of the village losing Internet connectivity for a short period. While onscreen content would be available during a period of Internet outage, residents would not be able to access the current screen grab. As a consequence we, the researchers, would have failed in our obligation with the village and the residents: they would feel a lack of control/awareness regarding the public face of the community, i.e., the content being shown on the public facing photo display. This can be considered as a problem residing at the community interaction design level.

The above example illustrates the impact of the hardware layer (unreliable internet connectivity in the village), impacting upon system architecture (need for web server/content source to be local rather than remote), further impacting on content (during internet outage the content would be available on the display but residents would not be able to remotely view the current screen grab), and finally having an effect on community interaction design (trust relationship between the researchers and residents).

Starting from the Hardware Layer. A good example comes from the FunSquare and Digifieds deployments: one of the display locations where observations were made was outdoors (in the city center). At that particular location, the sun created a lot of glare on the screen. This in turn made it hard for people to interact with any of the applications on the display. During the FunSquare observations, we noticed several instances where people pressed the '+' button repeatedly in order to see what would happen. However, because of the heavy glare they did not notice that the displayed facts changed. In other cases, people did not notice certain user interface elements, e.g., the timer. This shows how improper *hardware* can cause problems on *content* and *system interaction* layers. When these two are broken, it is much more difficult to stimulate *community interaction* through public displays.

Besides the display output qualities, some interventions also require on-screen input capabilities, i.e., touch screens. With today's prevalence of touch-enabled devices, touch is often seen as the default interaction modality. If the quality of a touch display does not meet user expectations (which are often high, since the majority of today's mobile phones typically feature high-resolution capacitive touch screens), it can have a significant negative impact upon user experience. For a highly interactive deployment such as the FunSquare game, we received comments that "the touch display is inaccurate, hard to use", that the game had "stiff controls", and that the overall experience with the game was "frustrating" or even "boring". In other words, the *hardware* had direct impact on *interaction* and *community interaction* layers.

One hardware issue that had an impact on system architecture directly in all three cases was unreliable Internet connectivity that created the need for *offline content access*. In the case of FunSquare (in both game and ambient mode), this meant having a stock of fun facts available for each display that would have been shown until the new/fresh ones arrived. In Digifieds, we did not manage to implement such offline content management in time, so the displays only worked in online mode, i.e., if there were problems with the Internet connection, no classifieds were available at all on the display (Digifieds that were created or retrieved using the mobile client were available offline). In the case of the Wray Photo Display this meant hosting the server locally on the Mac Mini running the display rather than at the university.

6 DISCUSSION

In this section, we show how the framework can be used practically for (1) a self-assessment exercise to calibrate one's individual awareness of the various issues involved in successful deployments (section 6.1). It can also be used to (2) tabulate encountered issues during a deployment in order to uncover follow-up issues (section 6.2). Also (3) different shapes of the framework can be used to better judge the necessary effort for each layer (section 6.3).

6.1 Using the P-LAYERS Framework to Self-Reflect on Individual Awareness and Interests

The framework can be used to calibrate one's individual awareness of the various issues involved in successful community deployments, given one's interests.

Two examples of such a reflective process are shown in Fig. 14: a researcher only interested in investigating how to create engaging community interaction design (shown on the left) might easily underestimate the amount of work that (a) needs to go into proper system interaction design, that (b) is required to provide sufficient supply of content, that (c) must ensure the robustness of the system, and (d) the amount of provisioning needed for supporting the hardware. As there is no inherent "up" or "down" in the layered framework, we can also reverse the order of the layers, as shown in the right-hand side of Fig. 14. This example shows a researcher interested in creating a robust system architecture for long-term deployments that address community engagement. While both the hardware and architecture might be in focus of the researcher, content and interaction issues might get overlooked.

In an iterative design process, such a self-assessment process can be used to repeatedly (e.g., after each iteration) self-reflect on one's individual level of awareness with respect to the challenges posed by the "hidden" layers, and to critically re-assess the amount of effort needed to address them.

6.2 Tabulating Issues to Uncover Follow-Up Issues

Alongside self-assessment, the P-LAYERS framework can be used in an iterative design process to document and uncover root causes and/or subsequent follow-up issues in application design, development, and deployment. Starting from a table (such as Table I), encountered or anticipated issues can be entered localized in a specific column or as a general issue in the leftmost column. Neighboring cells in the same row can then be brainstormed in order to uncover root causes or potentially related (and soon-to-be-encountered) issues in neighboring layers.

This can be helpful in an iterative deign process to predict possible issues at each layer. For example, in our Digifieds and FunSquare deployments we encountered several passers-by that avoided the public displays due to prior negative experiences with the system:

> "I don't use the displays because in my experience they don't work."

Such general lack of acceptance among the target community can be entered as a general issue in the leftmost column (cf. the first row of Table I) and then used to subsequently brainstorm potential causes on each of the layers: the content layer may show content that is inappropriate for the target community; the planned system interaction style might be unpopular with community members due to its embarrassing nature; or the community interaction design evokes negative connotations based on prior negative experiences with similar systems.

In the following, we provide a few additional examples from our deployment experiences and how they led to various rows in the table, allowing for both generalization of issues and subsequent envisioning of potentially related issues at other layers of the framework.

— *Changing Consumer Technology*. One issue that puts pressure on hardware selection for public display deployments "in the wild" is that it *ages fast*. Whereas in the lab it would be possible to quickly change the display that has the same look and feel as the trendiest/best technology currently available on the market, in the wild, once the displays are rolled out it is often very difficult to update them. As a consequence, after a while displays cannot compete with other devices with similar capabilities. Hence, they do not live up to user expectations. An example of this was evident in the FunSquare and the Digifieds deployments, where most users had smartphones with capacitive touch screens – which are much more responsive than the resistive ones used on the UBI Hotspots and also supported multi touch gestures. Consequently, most users where underwhelmed by the interaction experience with our applications as they expected the same experience on all devices. This is comparable to today's smartphone market, where people will typically blame the OS (e.g., Android) if the handset is not powerful enough to provide a smooth user experience.

— Another example of how changing consumer technology can lead to deployment issues are regional and/or demographic differences in technology uptake. In our Digifieds deployment, we used both a mobile website as well as a native Android OS application in order to take advantage of a fully integrated mobile phone OS. Although the assumption was that Android would be a commonly found OS in the environment (as indicated by its market share), to our surprise it was dwarfed by Symbian – a legacy OS with less than 1% market share[2] among smartphones. Yet in that particular setting it was the most used OS. An obvious solution is the use of cross-platform mobile phone solutions such as PhoneGap[3], though potentially at the expense of reduced platform functionality. Another case in point came in a local pilot deployment (unpublished) at the University of Lugano, where we envisioned students to upload photos from their mobile phone onto a public display. While this worked well in initial testing, the target event – the first week of classes – saw a large number of freshmen students from neighboring Italy who had disabled mobile services in Switzerland for fear of roaming charges, and who did not yet know about the university WiFi network. Consequently, uptake among freshmen was much lower than expected, significantly impacting the experience.

— *Changing third party services.* While the use of external services in our FunSquare deployment alleviated the need for generating sufficient amounts of content by users, it also introduced an additional dependency on third party services. We had to

[2] Nielsen: "Two Thirds of New Mobile Buyers Now Opting For Smartphones." Nielsen Blog, July 12, 2012. See http://blog.nielsen.com/nielsenwire/?p=32494

[3] http://phonegap.com/about

make sure that necessary content coming from these services, such as Facebook or Yahoo!Weather, remained available for fun fact generation, not only in light of our own unreliable Internet connectivity, but also in case of service outage. Similarly to our fact caching, we thus also stored every piece of intermediate information that was coming in from those services to generate our fun facts in order to counter connectivity interruptions. Although this lowered the risk of having "broken" content where one piece of information would have been missing, it is only a partial solution since it does not cover the case of a long-term service disruption. Ideally, one should ensure that each type of information could be sourced from a variety of third party services.

Working laterally along the different layers/cells, we can now easily envision other potential issues that may appear on other levels of the framework. For example, a café owner might decide to re-model its décor and decide that the display does not fit anymore. Or a free WiFi service that we have been relying on might start to move to a payment model.

— *Limitation of physical space*. Whenever a public display is installed it alters the physical space. This creates multiple issues. First, the display has to fit in with existing architecture. This can also create an issue on the hardware level as well, as a suitable display has to be found. On the system interaction level, this can limit the available choices of interaction. For example, if the only place where a display can be put is a high corner, this certainly rules out touch as interaction modality. Second, the interaction modality should not break the harmony between the space and people's activities, i.e., it should fit in with existing activities in a place. For example, while performing large gestures might be acceptable in places where people seek direct social interaction, e.g., alumni events [Rubegni et al. 2011], these would probably be seen as disruptive in a post office. In contrast, a touch-based interface suitable for a quiet village town hall might receive little attention at a noisy student reunion.

Once a public display application is deployed it starts competing with the environment. One example is systems where the application is assigned a fix time slot (time limitation). This was the case with FunSquare and Digifieds where both applications were hidden behind the commercial digital signage system and only became visible after users approached the display. In our case, this lead to an exposure time of less than 10% of the overall display time. Furthermore, public display applications also have to compete with other elements striving for attention in the environment. For example, while we conducted observations for FunSquare (ambient mode) a couple of street performers came to the statue, located close to the display and started their performance. People immediately started turning their heads and diverted their attention to the performers.

— *Professional support* can be crucial in many situations. For example in the case of the Wray Photo Display some of the early photos that were uploaded were taken by a professional. This helped to improve the initial user experience with the display and boost "buzz" around it. Similarly, a professional designed the FunSquare user interface and display template to achieve an attractive look. Professional help might also have been useful for the game design, as that would have had a significant impact on the community interaction experience.

We believe that in the future, professional support (graphics, game and interaction design, programmers) will become more important, also on other levels in the pyramid, in particular for the hardware and system architecture layers (e.g., for permanent outdoor installations such as the UBI hotspots). While all three of our applications were developed in the lab, every change requires testing in the real world setting. A simple example would be to test how the latest change affects the "in the wild" Internet connection (as in case of Wray Photo Display) or to check how the application performs on the hardware deployed in the real world setting

Table I. Use of the P-LAYERS framework for uncovering root causes and/or follow-up issues. Filling in a localized issue into the appropriate layer column helps brainstorming about how similar issues can affect other layers. Alternatively, a general issue can be entered in the leftmost column and be used to generate root causes in each of the layers.

	Layer				
Issue	Hardware	Sys. Arch.	Content	Sys. Interaction	CI Design
Lack of contextual acceptance			Inappropriate content	Embarrassing interaction style	Prior negative experiences
Changing consumer technology	Supporting new features/sensors, e.g., NFC, Kinect Fast aging hardware	Symbian, J2ME vs. Android, iOS	Content quality in contrast with what's on the market (e.g., 2MPX pics vs. 20MPX). Impacts, e.g., space needed to store photos.	System interaction broken down because of the file size of the new photos (larger photos = more space = more speed and bandwidth needed).	Wrong assumptions about availability of smartphones among target community, or lack of data plans for mobile connectivity
Changing 3rd party services	Free Wi-Fi suddenly become one that you have to pay for	Service replacement, or new service integrated	Disappearing content, e.g., weather service	Disappearing social media service that was used to gather user input to the service	Café closes or rearranges furniture. Display not appropriate anymore
Limitations of physical space	Influences the type of hardware to be used	Competing services	Too small font given viewer distance Short content glimpses to due movement speed of the passers-by	Fit with physical installation in the environment	Narrow hallway makes lingering in front of display socially awkward, as it blocks thoroughfare
Professional support	Maintenance	Buying licenses (e.g. middleware).	Content design	Hiring a professional interaction designer	Bringing in a community worker

(as in case of FunSquare and Digifieds). If the system is undergoing change during development cycles this leads to a need for constant revalidation "in the wild", i.e., the need to see how changes made in the lab work in the real world. This issue can be tackled by installing remote monitoring systems that eliminate (to a certain extent) the need to be physically present at the location and check how the change manifests. Alternatively, a designated "community worker" (or "Human Access Point" [Maunder et al. 2011]) could allow for a much more frequent in-situ evaluation, without having the need for the researchers to be in place, or the intrusive use of a camera.

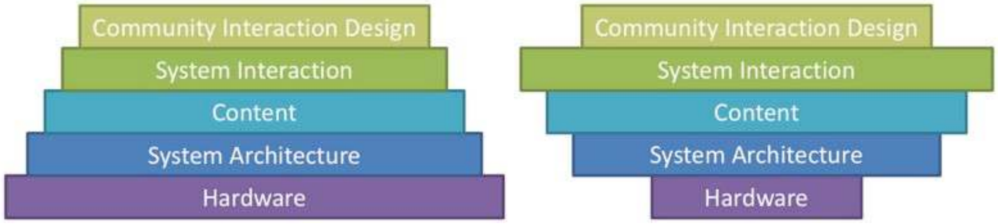

Fig. 15. Two different shapes of the P-LAYERS framework in order to incorporate expected effort – resources, costs, or time.

6.3 Using Shape as an Indicator of Effort

While the P-LAYERS framework is basically a set of layers, we considered a number of different shapes when drawing up the framework – starting with a pyramid, as shown in Fig. 15. In its current form, the framework simply offers the five layers, with no stance on any differences between the layers. However, when moving to a shape such as a pyramid, the framework can also be used for planning / to indicate effort, i.e., resources, cost, and time. *Resource* needs can come in many forms, e.g., necessary hardware for the deployment, third party system architecture, or people who would build the application. These resources have a certain *cost* associated with them. Cost in turn can take various forms: it can be monetary, e.g., cost of hardware or people; numerical, e.g., the number of people that are necessary to work on the system or the number of licenses necessary for a third party component; or it can include logistics costs, e.g., the overall cost of migrating from one hardware/software platform to another. We can look at resources also from the perspective of *time*, e.g., how much time is necessary to develop the community interaction/system interaction, or how much time is necessary to combine separately developed pieces together.

For example, if a research team is very much focused on creating the system architecture for community displays, they might in turn be underestimating the amount of effort needed for designing the actual community interaction around it. As depicted in Fig. 15, different shapes of the framework can be used to indicate the amount of effort invested in each layer. In an iterative process, change in the shape can be indicative of the effort for the particular iteration.

7 CONCLUSIONS AND FUTURE WORK

In this article, we have summarized our experiences in building, deploying, and evaluating public display applications that support community interaction “in the wild”. Informed by these experiences, we have presented the P-LAYERS framework comprising five layers, namely: 1) hardware, 2) system architecture, 3) content, 4) system interaction, and 5) community interaction design. The individual challenges and issues faced at each of the layers have been discussed, and illustrative crosscutting examples that show how the different layers interact presented.

Displays “in the wild” are special (though not unique) because of the strong interlinking between what are effectively engineering level and user experience issues and consequently the support required by research teams involved in the design and deployment of this class of system goes beyond that offered by more generic user experience guidelines. We believe that the P-LAYERS framework can provide useful support for researchers/practitioners by alerting them to the diverse range of issues they are likely to encounter with the development, deployment, and maintenance of public display systems that aim at simulating community interaction. Taking just one very practical example, research teams should be careful to ensure that their test hardware matches the hardware deployed 'in the wild' as differences can adversely affect system interaction. This example also illustrates another purpose of our framework, namely providing researchers with an appreciation of the impact that problems arising from one specific layer can have on other layers.

Research teams can also use the framework to gain a systematic understanding of these different layers of concern and this, in turn, should help them to plan and allocate resources appropriately. Furthermore, the framework can help a research team to consider how a given issue, e.g., professional support, can have an impact/implication in each different layer. In future work, we would seek to further validate the framework, firstly through comparison to the experiences of other documented public display deployments and secondly through new deployments that utilize our framework throughout the development process. We also believe that the framework could be used as part of iterative, participatory design processes. In particular, we would explore the use of P-LAYERS as a boundary object [Star and Griesemer 1989] to improve communication and understanding of deployment issues between researchers, participants and other collaborators. This might be achieved, for example, by exposing participants to the multi-faceted issues surrounding public displays while remaining focused on the particular aspects of the framework that are most critical to the deployment.

ACKNOWLEDGMENTS

The research leading to these results has received funding from the European Union Seventh Framework Programme (FP7/2007-2013) under grant agreement no. 244011. The research related to the Wray Photo Display was funded through the EPSRC EPSRC CASIDE (grant number EP/C005589/1) and SHARC (grant number EP/K015850/1) projects and a Microsoft Research European PhD scholarship.

REFERENCES

ALT, F., MEMAROVIC, N., ELHART, I., BIAL, D., SCHMIDT, A., LANGHEINRICH, M., HARBOE, G., HUANG, E., AND SCIPIONI, M.P . 2011a. Designing shared public display networks – implications from today's paper-based notice areas. In *Proceedings of the 9th International Conference on Pervasive Computing (Pervasive'11).* Springer, Berlin, Heidelberg, 258–275. DOI:10.1007/978-3-642-21726-5_17

ALT, F., KUBITZA, T., BIAL, D., ZAIDAN, F., ORTEL, M., ZURMAAR, B., LEWEN, T., SHIRAZI, A.S., AND SCHMIDT, A. 2011b. Digifieds: Insights into Deploying Digital Public Notice Areas in the Wild. In *Proceedings of the 10th International Conference on Mobile and Ubiquitous Multimedia (MUM'11).* ACM, New York, NY, 165–174. DOI: 10.1145/2107596.2107618

ALT, F., SAHAMI SHIRAZI, A., KUBITZA, T., SCHMIDT, A . 2013. Interaction Techniques for Creating and Exchanging Content with Public Displays. In *Proceedings of the 31st international conference on Human factors in computing systems* (*CHI'13).* ACM, New York, NY (in press).

CHALMERS, M., AND MACCOLL, I. 2003. Seamful and seamless design in ubiquitous computing. In *Workshop At the Crossroads: The Interaction of HCI and Systems Issues in UbiComp*. URL: http://bit.ly/Vvr6j4

CHEVERST, K, CLARKE, K, FITTON, D, ROUNCEFIELD, M, CRABTREE, A, AND HEMMINGS, T. 2002. SPAM on the menu: the practical use of remote messaging in community care. *SIGCAPH Comput. Phys. Handicap.* 73-74 (June 2002), 23-29. DOI:10.1145/960201.957210

CHEVERST, K. DIX, A. FITTON, D., KRAY, C., ROUNCEFIELD, M., SAS, C., SASLIS-LAGOUDAKIS, G. AND SHERIDAN, G. 2005. Exploring bluetooth based mobile phone interaction with the hermes photo display. In *Proceedings of the 7th international conference on Human computer interaction with mobile devices & services* (MobileHCI '05). ACM, New York, NY, 47-54. DOI:10.1145/1085777.1085786

CHEVERST, K., TAYLOR, N., ROUNCEFIELD, M., GALANI, A., AND KRAY, C. 2008. The Challenge of Evaluating Situated Display based Technology Interventions Designed to Foster 'Sense of Community'. In *Proceedings of workshop on Ubiquitous systems evaluation (USE'08),* held in conjunction with Ubicomp'08.

CHURCHILL, E.F., NELSON, L., AND DENOUE, L . 2003. Multimedia fliers: information sharing with digital community bulletin boards. In *Communities and technologies (C&T'03).* Kluwer, B.V., Deventer, The Netherlands, The Netherlands 97-117. DOI:10.1007/978-94-017-0115-0_6

CHURCHILL, E.F., NELSON, L., AND HSIEH, G. 2006. Café life in the digital age: augmenting information flow in a café-work-entertainment space. In *CHI '06 Extended Abstracts on Human Factors in Computing Systems* (CHI EA '06). ACM, New York, NY, 123-128. DOI:10.1145/1125451.1125481

CLARK, A. 2007. Understanding Community: A review of networks, ties and contacts. Available at: http://eprints.ncrm.ac.uk/469/ [Accessed June 23, 2012]

CLINCH, S., DAVIES, N., FRIDAY, A., AND EFSTRATIOU, C. 2011. Reflections on the long-term use of an experimental digital signage system. In *Proceedings of the 13th international conference on Ubiquitous computing* (UbiComp '11). ACM, New York, NY, 133-142. DOI:10.1145/2030112.2030132

DAVIES, N., LANGHEINRICH, M., JOSÉ, R., AND SCHMIDT, A . 2012. Open Display Networks: A Communications Medium for the 21st Century. *IEEE Computer.* 45, 5 (May 2012), 58–64. DOI:10.1109/MC.2012.114

FATAH GEN SCHIECK, A., KOSTAKOS,V. AND PENN, A. 2010. Exploring Digital Encounters in the Public Arena. In *Shared Encounters*, WILLIS, K.S., ROUSSOS, G., CHORIANOPOULOS AND STRUPPEK, M. Eds. Springer, London, 179-195. DOI:10.1007/978-1-84882-727-1_9

FISCHER, G. 2001. Communities of interest: Learning through the interaction of multiple knowledge systems. In *Proceedings of 24th Annual Information Systems Research Seminar In Scandinavia (IRIS'24),* 1– 4.

HILLERY, G.A. 1955. Definitions of community: areas of agreement. *Rural Sociology, 20*, 111-123.

HINRICHS, U., SCHMIDT, H., AND CARPENDALE. S . 2008. Emdialog: Bringing information visualization into the museum. *IEEE Trans. on Vis. and Com. Graph.* 14, 6 (Nov.-Dec. 2008), 1181–1188. DOI: 10.1109/TVCG.2008.127

HOUDE, S., BELLAMY, R., AND LEAHY, L. 1998. In search of design principles for tools and practices to support communication within a learning community. *SIGCHI Bull.* 30, 2 (April 1998), 113-118. DOI:10.1145/279044.279171

HUANG, E. M., KOSTER, A,. AND BORCHERS, J. 2008. Overcoming assumptions and uncovering practices: When does the public really look at public displays? In *Proceedings of the 6th International Conference on Pervasive Computing (Pervasive '08).* Springer-Verlag, Berlin, Heidelberg, 228-243. DOI:10.1007/978-3-540-79576-6_14

IZADI, S., FITZPATRICK, G., RODDEN, T., BRIGNULL, H., ROGERS, Y., AND LINDLEY, S. 2005. The iterative design and study of a large display for shared and sociable spaces. In Proceedings of the 2005 conference on Designing for User eXperience (DUX '05). AIGA: American Institute of Graphic Arts, New York, NY, Article 59.

JACUCCI, G., MORRISON, A., RICHARD, G.T., KLEIMOLA, J., PELTONEN, P., PARISI, L., AND LAITINEN. T. 2010. Worlds of information: designing for engagement at a public multi-touch display. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (CHI '10). ACM, New York, NY, 2267-2276. DOI:10.1145/1753326.1753669

JONES, M., HARWOOD, W., BAINBRIDGE, D., BUCHANAN, G., FROHLICH, D., RACHOVIDES, D., FRANK, M., AND LALMAS. M. 2008. *"Narrowcast yourself"*: designing for community storytelling in a rural Indian context. In *Proceedings of the 7th ACM conference on Designing interactive systems* (DIS '08). ACM, New York, NY, 369-378. DOI:10.1145/1394445.1394485

JOSÉ, R., PINTO, H., SILVA, B., MELRO, A., AND RODRIGUES, H. 2012. Beyond interaction: tools and practices for situated publication in display networks. In *Proceedings of the 2012 International Symposium on Pervasive Displays* (PerDis '12). ACM, New York, NY, Article 8, 6 pages. DOI:10.1145/2307798.2307806

LANGHEINRICH, M., MEMAROVIC, N., ELHART, I., AND ALT, F. 2011. Autopoiesic Content: A Conceptual Model for Enabling Situated Self-generative Content for Public Displays. In 1st International Workshop on Pervasive Urban Applications (San Francisco, CA, USA, June 12, 2011).

LÉVESQUE, M., BÉLANGER, L., AND LEWIS, J. 2006. p2P: Cityspeak's Reconfiguration of Public Media Space. *Wi: J. Mob. Digi. Comm. Net.* l, 1.

MARSHALL, P., MORRIS, R., ROGERS, Y., KREITMAYER, S., AND DAVIES, M. 2011. Rethinking 'multi-user': an in-the-wild study of how groups approach a walk-up-and-use tabletop interface. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (CHI '11). ACM, New York, NY, 3033-3042. DOI:10.1145/1978942.1979392

MATURANA, H. AND VARELA, F. 1980. Autopoiesis and Cognition: the Realization of the Living. *Boston Studies in the Philosophy of Science* 42, 1.

MAUNDER, A. MARSDEN, G. AND HARPER, R. 2011. Making the link — providing mobile media for novice communities in the developing world. *Int. J. Hum.–Comput. St.*, 69, 10 (Sep. 2011), 647–657. DOI:10.1016/j.ijhcs.2010.12.009

MCCARTHY, J.F., COSTA, T.J., AND LIONGOSARI, E.S. 2001. UniCast, OutCast & GroupCast: Three Steps Toward Ubiquitous, Peripheral Displays. In *Proceedings of the 3rd international conference on Ubiquitous Computing (UbiComp '01).* Springer-Verlag, London, UK, 332-345. DOI:10.1007/3-540-45427-6_28

MCCARTHY, J.F., FARNHAM, S.D., PATEL, Y., AHUJA, S., NORMAN, D., HAZLEWOOD, W.R., AND LIND. J. 2009. Supporting community in third places with situated social software. In *Proceedings of the fourth international conference on Communities and technologies* (C&T '09). ACM, New York, NY, 225-234. DOI:10.1145/1556460.1556493

MCDONALD, D.W., MCCARTHY, J.F., SOROCZAK, S., NGUYEN, D.H., AND RASHID. A.M. 2008. Proactive displays: Supporting awareness in fluid social environments. *ACM Trans. Comput.-Hum. Interact.* 14, 4, Article 16 (Jan. 2008), 31 pages. DOI:10.1145/1314683.1314684

MCMILLAN, D.W. AND CHAVIS, D.M . 1986. Sense of community: A definition and theory. *Journal of community psychology*. 14, 1, 6 – 23.

MEMAROVIC, N., ELHART, I., AND LANGHEINRICH, M. 2011. FunSquare: first experiences with autopoiesic content. In *Proceedings of the 10th International Conference on Mobile and Ubiquitous Multimedia* (MUM '11). ACM, New York, NY, 175-184. DOI:10.1145/2107596.2107619

MEMAROVIC, N., AND LANGEHIRNICH, M . 2010. "Your place or mine?" - Connecting communities and public places through networked public displays. In Urban IOT Workshop at IOT 2010.

MEMAROVIC, N., LANGHEINRICH, M., AND ALT, F. 2011a. Connecting people through content – Promoting community identity cognition through people and places. In *Proceedings of Community Informatics 2011*.

MEMAROVIC, N., LANGHEINRICH, M., AND ALT, F. 2012. The interacting places framework: conceptualizing public display applications that promote community interaction and place awareness. In *Proceedings of the 2012 International Symposium on Pervasive Displays* (PerDis '12). ACM, New York, NY, Article 7, 6 pages. DOI:10.1145/2307798.2307805

MEMAROVIC, N., LANGHEINRICH, M., ALT, F., ELHART, I., HOSIO, S., AND RUBEGNI, E. 2012. Using public displays to stimulate passive engagement, active engagement, and discovery in public spaces. In *Proceedings of the Media Architecture Biennale Conference: Participation* (MAB '12). ACM, New York, NY, 55-64. DOI:10.1145/2421076.2421086

MEMAROVIC, N., LANGHEINRICH, M., RUBEGNI, E., DAVID, A., AND ELHART, I. 2012. Designing "interacting places" for a student community using a communicative ecology approach. In *Proceedings of the 11th International Conference on Mobile and Ubiquitous Multimedia* (MUM '12). ACM, New York, NY, Article 43, 10 pages. DOI:10.1145/2406367.2406420

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 *Proceedings of the 7th International Conference on Pervasive Computing (Pervasive '09).* Springer-Verlag, Berlin, Heidelberg, 1-8. DOI:10.1007/978-3-642-01516-8_1

MUNSON, S.A., ROSENGREN, E., AND RESNICK, P. 2011. Thanks and tweets: comparing two public displays. In *Proceedings of the ACM 2011 conference on Computer supported cooperative work* (CSCW '11). ACM, New York, NY, 331-340. DOI:10.1145/1958824.1958875

MYNATT, E.D., ADLER, A., ITO, M., AND O'DAY, V.L. 1997. Design for network communities. In *Proceedings of the ACM SIGCHI Conference on Human factors in computing systems* (CHI '97). ACM, New York, NY, 210-217. DOI:10.1145/258549.258707

O'HARA, K., PERRY, M., AND LEWIS, S. 2003. Social coordination around a situated display appliance. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (CHI '03). ACM, New York, NY, 65-72. DOI:10.1145/642611.642624

OJALA, T., AND KOSTAKOS, V. 2011. UBI challenge: research coopetition on real-world urban computing. In *Proceedings of the 10th International Conference on Mobile and Ubiquitous Multimedia* (MUM '11). ACM, New York, NY, 205-208. DOI:10.1145/2107596.2107622

OJALA, T., KOSTAKOS, V., AND KUKKA, H. 2011. It's a Jungle Out There: Fantasy and Reality of Evaluating Public Displays "in the Wild". In *Proceedings of workshop on Large Displays in Urban Life - from Exhibition Halls to Media Facades*, held in conjunction with CHI 2011.

OJALA, T., KOSTAKOS, V., KUKKA, H.,HEIKKINEN, T., LINDÉN, T., JURMU, M., HOSIO, S., KRUGER, F., AND ZANNI, D. 2012. Multipurpose Interactive Public Displays in the Wild: Three Years Later. *IEEE Compt.* 45, 5 (May 2012), 42 – 48. DOI:10.1109/MC.2012.115

OJALA, T., KUKKA, H., LINDÉN, T., HEIKKINEN, T., JURMU, M., HOSIO, S., AND KRUGER, F. 2010. UBI-hotspot 1.0: Large-scale Long-term Deployment of Interactive Public Displays in a City Center. In *Proceedings of 2010 5th International Conference on Internet and Web Applications and Services (ICIW'10).* IEEE CPS, 285 - 294. DOI:10.1109/ICIW.2010.49

PELTONEN, P., KURVINEN, E., SALOVAARA, A., JACUCCI, G., ILMONEN, T., EVANS, J., OULASVIRTA, A., AND SAARIKKO, P. 2008. It's Mine, Don't Touch!: interactions at a large multi-touch display in a city centre. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (CHI '08). ACM, New York, NY, 1285-1294. DOI:10.1145/1357054.1357255

RAMSEY, D., AND BEESLEY, K. 2007. 'Perimeteritis' and rural health in Manitoba, Canada: perspectives from rural healthcare managers. *Rural and Remote Health* 7, 850. Available: http://www.rrh.org.au

REDHEAD, F., AND BRERETON. M. 2009. Designing interaction for local communications: An urban screen study. In *Proceedings of the 12th IFIP TC 13 International Conference on Human-Computer Interaction: Part II (INTERACT '09).* Springer-Verlag, Berlin, Heidelberg, 457-460. DOI:10.1007/978-3-642-03658-3_49

RUBEGNI, E., MEMAROVIC, N., AND LANGHEINRICH, M. 2011. Talking to Strangers: Using Large Public Displays to Facilitate Social Interaction. In *Proceedings* of *14th International Conference on Human-Computer Interaction (HCII 2011),* Springer, 195-204. DOI:10.1007/978-3-642-21708-1_23

SCHROETER, R. 2012. Engaging new digital locals with interactive urban screens to collaboratively improve the city. In *Proceedings of the ACM 2012 conference on Computer Supported Cooperative Work* (CSCW '12). ACM, New York, NY, 227-236. DOI:10.1145/2145204.2145239

SCHROETER, R, MARCUS, F., AND CHRISTINE, S. 2012. People, content, location: sweet spotting urban screens for situated engagement. In *Proceedings of the Designing Interactive Systems Conference* (DIS '12). ACM, New York, NY, 146-155. DOI:10.1145/2317956.2317980

SNOWDON, D., AND GRASSO, A. 2002. Diffusing information in organizational settings: learning from experience. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (CHI '02). ACM, New York, NY, 331-338. DOI:10.1145/503376.503435

STAR, S. AND GRIESEMER, J. 1989. Institutional Ecology, 'Translations' and Boundary Objects: Amateurs and Professionals in Berkeley's Museum of Vertebrate Zoology, 1907–39. *Social Studies of Science.* 19, 3, 387–420. DOI:10.1177/030631289019003001

STORZ, O., FRIDAY, A. DAVIES, N., FINNEY, J., SAS, C., AND SHERIDAN, J. 2006. Public Ubiquitous Computing Systems: Lessons from the e-Campus Display Deployments. *IEEE Perv. Comp.* 5, 3 (July 2006), 40-47. DOI:10.1109/MPRV.2006.56

TAYLOR, N., AND CHEVERST, K. 2009. Social interaction around a rural community photo display. *Int. J. of Hum.-Comp. Studies*, 67, 12 (Dec. 2009), 1037–1047. DOI:10.1016/j.ijhcs.2009.07.006

TAYLOR, N., AND CHEVERST, K. 2010. Creating a rural community display with local engagement. In *Proceedings of the 8th ACM Conference on Designing Interactive Systems* (DIS '10). ACM, New York, NY, 218-227. DOI:10.1145/1858171.1858209

TAYLOR, N., AND CHEVERST, K. 2012. Supporting Community Awareness with Interactive Displays. *IEEE Compt.* 45, 5 (May 2012), 26 – 32.

TAYLOR, N., CHEVERST, K., WRIGHT, P. AND OLIVIER, P. 2013. Leaving the Wild: Lessons from Community Technology Handovers**.** In *Proc. of the ACM International Conference on Computer Human Interaction (CHI'2013).* ACM, New York, NY (in press).

WENGER, E. 1998. *Communities of practice: Learning, meaning, and identity*. Cambridge University Press, Cambridge, UK.

WHYTE, W. H. 1980. *The Social Life of Small Urban Spaces*. Project for Public Spaces. The Conservation Foundation, Washington, DC.