Towards a Logging Framework for Evaluation and Management of Information Radiators

Michael Koch
Department of Informatics
Bundeswehr University Munich
Munich, Germany
michael.koch@unibw.de

ABSTRACT

In the CommunityMirror project we are researching networked information radiators that are based on large interactive screens, showing information from different sources. For addressing our constructive research questions – e.g. to find design parameters, to find solutions for implementing features and processes - we are facing different problems acquiring the information about usage and non-usage in long-term deployments. While working on the issues we found an interesting parallel: Most of the information needed for scientific evaluation is also needed for (platform) management issues. In this paper we briefly present the problem scenario – and the current concept for addressing this issue in the CommunityMirror implementation – in the form of a concept for a logging framework.

KEYWORDS

CommunityMirror, Evaluation, Large Screen, (Semi-)Public Display, Information Radiator, Management, Logging Framework

Reference format:

Michael Koch (2019): Towards a Logging Framework for Evaluation and Management of Information Radiators. In *Mensch und Computer 2019 – Workshopband*, Bonn: Gesellschaft für Informatik e.V., https://doi.org/10.18420/muc2019-ws04-566

1 (Networked) Information Radiators and the CommunityMirror Project

The term “information radiator” has first been coined by Alistair Cockburn for frequently updated posters showing the current state in software development processes in a high traffic hallway [5]. The radiators provide pieces of information or in other words visual representations of information objects stored in the underlying data sources in a way that makes them consumable peripherally. In contrast to most other IT systems which only “provide” information after a certain user interaction (e.g. a search) information radiators proactively distribute their “info particles” independently from any user in order to generate appreciation for the contributors and thereby motivate them for further participation and sharing [18].

Some examples of research prototypes exploring the design space of interactive information radiators over the past ten years are CommunityWall [28], Plasma Poster Network [4], PD-Net [12], Ambient Surfaces [24,25], XioScreen [7] and CommunityMirror [9,17,18,20]. An example of a public deployment with long-term evaluation can be found in the UBI-Hotspots project [16].

Figure 1. CommunityMirror in use [18].

Following the idea of information radiators, CommunityMirrors are multi-user multi-touch information applications for semi-public or public use [9,17,18,20]. They support several co-located simultaneously interacting users. In idle mode without users in interacting with the displays CommunityMirrors show an animated flow of info particles on the screen. Touching an info particle opens a more detailed view of the particle and shows related particles as a visual graph that can be explored by the user. Additionally, static (sticky) components are shown on the screen, and occasionally, info particles are displayed in a large

MuC Workshops 2019, Hamburg, Germany

https://doi.org/10.18420/muc2019-ws-566

teaser view that covers half of the screen (comparable to wall posters).

Several CommunityMirror instances that show information from the same information pool form a set of networked information radiators. In addition to (context/location-sensitively) displaying information from the same pool, the devices can also provide different media space functionality for connecting the users in front of the different screens [14].

In the past ten years, different CommunityMirror applications have been built for different usage settings. For example the application was used to provide an interactive participant list for (academic) conferences, an awareness display for team spaces or a person-centric catalogue for exhibitions [9].

In the different long-term real-world deployments we experimented what information on the usage can be collected and what information should be collected to answer our research questions (see Section 2). This position paper summarizes the lessons we learned in this context and derives a concept for a logging framework for our CommunityMirror software from these learnings.

Related work on methods used for evaluating long-term deployments of ambient displays can be found in reports about single research projects or papers summarizing the learnings from single projects [2,11,12,15,21,23,27].

2 Research Questions and Interesting Information for Addressing these Questions

In the past years of the CommunityMirror project, we found different information of particular interest for the evaluation of different design parameters – particularly information about passive usage and information about (usage) context. In this section we try to summarize what we learned about needs and possibilities – for having a list of requirements for the design of the logging framework. Thereby, we focus on long-term real-world deployments with different forms of observation (of users / usage) including interviews and questionnaires.

2.1 Research Questions

To motivate which information we would like to collect, we first will briefly address which research questions we want address in the context of networked information radiators.

In general, we are trying to find out about how such (networked) information radiators can and should be designed and set up to provide benefit to the users (and the organization employing the users). This includes working out and evaluating design recommendations, and to find out more about different effects that can be observed during the operation of such devices/applications (e.g. the novelty effect [8] or the honey pot effect [3]) – and on how to address these effects in evaluation or make use of the effects in operating the information radiators.

The design parameters, we would like to have well-founded proposals for, are:

- What information to display? (What information is of interest (for which target groups)? What information is useful? …)
- How to display the information (layout, animation, usage of images, usage of catchy headlines, display of related information, …) – to allow easy reception and to attract attention?
- What interaction possibilities to offer to the user? (How does interactivity influence usage / benefit? Which kind of interactivity is most beneficial [29].)
- Where to install the information screens?

Additionally, we are interested in how to obtain the information in the needed quality (with as little as possible additional effort) – more general: how to operate and administrate a networked information radiator.

In all these questions we want to better understand basic workings – e.g. by creating models that allow a better understanding of usage patterns, user motivation etc. - and create design recommendations in different forms.

For all of these goals it is important to monitor usage over larger periods of time and to compare different settings. It is important to have aggregated information over longer periods of time and to be able to look into single instances in analysis.

So, we would like to observe usage in long-term real-life deployments – particularly

- How do users use the screen?
- What is the benefit the users perceive?
- How do users interact with each other?

2.2 "Classical" interaction information

The first thing that comes to one's mind when thinking about what information should be captured automatically in long-term deployment is active usage information (information about touch interaction with the screen). This type of information is available easily.

When trying to use this information to derive if the right information has been displayed on the screen or if there is too much or too little information on the screen, we learned that we need very rich context information together with the touch events. For example, the event of a user touching and dragging an info particle could be interpreted directly as showing interest in that particle. However, something that we have repeatedly observed in the wild is users moving a particle out of the way in order to read the one displayed below it. This kind of rich context information would need to be available in the logs so the full picture of the interaction can be understood after the fact. Ideally, we would have a functionality to playback the contents and animations on the screen at a particular point in time – useful enough would be to capture screen shots at the interaction moments. The best way how to do this is still open.

2.3 Information about passive usage

In observation studies we found that people often benefitted from the information on the screen without interacting with it – which was not captured in interaction logs.

Additionally, we found it hard to evaluate the interaction logs, because we did not know how many potential users there were in front of the screen during the longer periods of time.

So, we would like to have information on:

- How many people are walking by the installation?
- Does the installation change their activity – e.g. slow them down, make them stop?
- Do they look at the installation?
- Do they approach the installation?

One hypothesis in regard to CommunityMirrors is that the installations initiate interaction between people in front of the screen. To evaluate this, it would be interesting to have information about groups and about what is happening in the groups.

Since it is impractical to obtain this information through observation (both direct observation and recording and manually analyzing videos of the scene) in long-term deployments, we need a solution for automatically capturing the information.

Our approach to this is to use tools like the Microsoft Kinect that automatically captures information from the video feed. In case of the Kinect we experimented with a logging solution from Jan Schwarzer that records skeleton information (.xef file format) for intervals where persons are recognized in front of the screen, and extracts number of persons, number of positive engaged values, number of negative engaged values and minimal distance of the person that came next to the screen.

Related work: [6][30] both using video / Kinect feeds.

2.4 Information about context

In the past section we reported that direct context information is needed to interpret interaction events – e.g. information about what type of information is displayed and if it includes an image to interpret a touch event.

In addition to the screen context, we would also like to have information about the direct usage context – i.e. if the user was alone in the front of the screen, if the user was part of a group (how large?), if people were standing nearby and watching. This information could also be recorded using video analysis tools as presented in the previous section.

When looking at usage and non-usage data over long periods of time we additionally found the need for information about the larger context of the usage situation – e.g. about what has been happening in the organization where the device was used – e.g. if it was a bank holiday, if there was a special event nearby, … (see also [26]).

While information about active and passive usage and about the people standing nearby can be made available via sensors, this kind of information is hard to capture in a logging framework. Here we need to consult (or automatically import) calendars and similar external data sources. We are currently trying to capture such information by hand in experiment diaries and use them manually when analyzing data. But we would like to store and capture them in a more structured form to allow automatic analysis. This might be too much for a logging framework – but at least the possibility to store such information in the logs (in the form of a structured experiment diary) should be supported.

3 Information for Administrating Networked Information Radiators

In our different long-term deployments, we faced issues of managing the deployment – which is first the management or technical administration of the installation itself, and more important the management of the content, i.e. acquiring and annotating content for being displayed on the information radiators.

Content management is about keeping an overview of available content – and of available quality of content (e.g. availability of images, of short titles etc.) – and of providing means for easily inserting and editing the content.

This is due to the learning that some governing has to be done – completely automated harvesting of content from other sources did not work – for example see the learnings in [19].

So, there has to be the functionality of a CMS and of a dashboard for helping the administrators to decide if there is enough (interesting) content available.

Interestingly, some the information needed in a dashboard for governance is quite similar to what is needed for scientific evaluation: we need information about what content is of interest to the users (being interacted with, being looked at).

So, in addition to requirements from the research questions to address with the deployments the requirements for the CommunityMirror logging framework should also include the requirements administrators have for running an information radiator network. This field still has to be researched in more detail.

Related work in this context can be found in dashboards and dedicated CMS solutions for digital signage, e.g. Magicinfo from Samsung, Supersign from LG, Broadsign, Youdeck, Scala, viewne and more.

4 Logging Framework for Networked Information Radiators

In this section we briefly present the current version of our logging framework for the CommunityMirror framework.

4.1 Data model

For presenting the logging framework, we first have to detail the data objects that are processed in the CommunityMirror applications.

First, we distinguish between information objects and representations of the information objects on the screen. Information objects are managed in a service called CommunityMashup [10] and have a unique identifier to refer to (and to use for retrieving the information later when evaluating the deployment). Information objects on screen are called info particle (or visual information object) and always include a link to the information object they visualize.

On the screen we distinguish

- Views (complete screens or independent areas covering a part of a screen potentially showing several objects)
- Single visual information objects (info particles)

Info particles can be displayed in different levels of detail – the simplest version providing a micro view (just showing a picture / title) and a detail view (showing details). In the detail view, connections to other info particles (and the corresponding information objects) are shown in the form of a navigable graph.

There is a standard view – the so-called flow view that displays a flow of info particles plus some static information/interaction objects showing external information (e.g. bus schedules) or triggering something on the screen (buttons for changing views, for influencing the display).

All visual objects have unique identifiers for allowing reference during evaluation.

4.2 Information classes

Summarizing the needs from evaluation and from administration, we collected the following requirements of information classes for a logging framework:

- Direct interaction (touch events, drag events, resize events, rotate events) – classical information like screen coordinates – information about (information) objects that have been touched. – Class: user-activity
- Activities of users in front of the screen that are not interacting with the screen. Information about the time period the user were present, about the interaction zone the users were in and if possible about where the user looked. – Class: user-passive
- Information about what has been displayed on the screen or removed from the screen when people have interacted with it (to reproduce which objects have been on the screen at a particular time) – could be implemented by screen images (in interaction relevant times). Class: display-snapshot, display-show, display-hide.
- Information about what is happening in the surroundings of the screen – what people are standing there, where they are standing (interaction zones [1,13,18,22]), what people are moving. Class: surroundings-snapshot, surroundings-arrive, surroundings-leave
- Information about data that could be shown on the screen (for governance). Class: data-available, data-loaded
- Information about the direct environment of a screen (people in from of the screen, …). Class: environment
- Information about the extended environment / context (opening hours of the building, holidays, outages of the hardware, events taking place near the screen) – this also might include free text comments by the administrators. This information class also can be used to record results of interviews or user questionnaires. – Class: comment

Log data entries can relate to a point in time or to a period of time.

In addition to simple log data entries, it would be beneficial to aggregate log data entries – e.g. to sessions – and to make the information on the sessions available for evaluation (or administration). However, this issue is not trivial. There are different ways to define, what a session is – and it should be possible during evaluation to try different ways to do the aggregation. So, the information about sessions probably should not be stored with the log data entries directly. This question is still open.

4.3 Information class attributes

Summarizing the findings of different evaluation tasks, we identified the following general attributes to be stored with all kinds of log data entries:

- screen-id (to distinguish different screens in a multi-screen / networked screen setup)
- view-id
- (linked) information object-id (if there is any)
- user-id (if the (interacting) user has been identified in some way)
- timestamp-start
- timestamp-end
- data type/class (which kind of log data entry – see Section 4.2)

Information about the screen context (what was happening on the screen) and about the surroundings context (how many people or groups are nearby) can either be stored as attributes or be stored as separate log data entries – and be linked by the timestamp.

For the different data classes, we identified additional attributes that have already been briefly mentioned in the list of data classes in the previous section.

5 Summary and Outlook

In this position paper we collected some experience and thoughts on what information to collect in long-term evaluations of (semi-)public display deployments – and what to use this information for.

We currently see two sets of questions to address in out further work:

1) Questions regarding the design of the logging solution – e.g. about how to track information about passive usage and context information – this includes both technical design and designing processes around the interfaces
2) Questions regarding the usage of the log data – e.g. how to visualize the data for making it useful for researchers and administrators (e.g related work [31]).

We will continue our work in implementing the described logging framework in the next version of our CommunityMirror software – and put it into use.

Primary focus currently is on capturing information about passive usage and about the surroundings context (people near the screen) – and on visualization of the information for researchers and for administrators (developing an administration dashboard).

REFERENCES

1. Florian Alt, Andreas Bulling, Lukas Mecke, and Daniel Buschek. 2016. Attention, please! Comparing Features for Measuring Audience Attention Towards Pervasive Displays. In *Proc. Designing Interactive Systems (DIS).*
2. Florian Alt, Stefan Schneegaß, Albrecht Schmidt, Jörg Müller, and Nemanja Memarovic. 2012. How to evaluate public displays. In *Proceedings of the 2012 International Symposium on Pervasive Displays (PerDis 2012)*, 1–6. https://doi.org/10.1145/2307798.2307815
3. Harry Brignull and Yvonne Rogers. 2003. Enticing People to Interact with Large Public Displays in Public Spaces. In *Proc. IFIP TC13 International Conference on Human-Computer Interaction (INTERACT)*, 17–24.
4. Elizabeth F Churchill, Les Nelson, L Denoue, Paul Murphy, and J Helfman. 2003. The Plasma Poster Network. In *Public and Situated Displays - Social and Interactional Aspects of Shared Display Technologies*, Kenton O'Hara, Ethan Perry, Elizabeth F Churchill and Daniel M Russel (eds.). 233–260.
5. Alistair Cockburn. 2008. Information Radiator. Retrieved from http://alistair.cockburn.us/Information+radiator
6. Ivan Elhart, Mateusz Mikusz, Cristian Gomez Mora, Marc Langheinrich, and Nigel Davies. 2017. Audience monitor: an open source tool for tracking audience mobility in front of pervasive displays. In *PerDis '17 Proceedings of the 6th ACM International Symposium on Pervasive Displays*. https://doi.org/10.1145/3078810.3078823
7. Ludwig John and Thomas Rist. 2012. xioScreen: Experiences Gained from Building a Series of Prototypes of Interactive Public Displays. In *Ubiquitous Display Environments*, Antonio Krüger and Tsvi Kuflik (eds.). Springer, Berlin, 125–142.
8. Michael Koch, Kai von Luck, Jan Schwarzer, and Susanne Draheim. 2018. The Novelty Effect in Large Display Deployments – Experiences and Lessons-Learned for Evaluating Prototypes. In *Proc. 16th European Conference on Computer-Supported Cooperative Work.* https://doi.org/10.18420/ecscw2018_3
9. Michael Koch, Florian Ott, and Alexander Richter. 2014. Socio-technically Integrated Access to Virtual Communities with CommunityMirrors. In *Virtual Communities*, Jan Marco Leimeister and Balaji Rajagopolan (eds.). M. E. Sharpe, Armonk, NY, 145–161.
10. Peter Lachenmaier, Florian Ott, and Michael Koch. 2013. Model-driven development of a person-centric mashup for social software. *Social Network Analysis and Mining* 3, 2: 193–207. https://doi.org/10.1007/s13278-012-0064-x
11. Tara Matthews, Gary Hsieh, and Jennifer Mankoff. 2009. Evaluating Periperal Displays. In *Awareness Systems*, Panos Markopoulos, Boris de Ruyter and Wendy E MacKay (eds.). Springer, London, 447–472.
12. Nemanja Memarovic, Ivan Elhart, and Elisa Rubegni. 2016. Developing a Networked Public Display System. *IEEE Pervasive Computing* 15, 3: 32–39. https://doi.org/10.1109/MPRV.2016.59
13. Jörg Müller, Florian Alt, Daniel Michelis, and Albrecht Schmidt. 2010. Requirements and design space for interactive public displays. In *Proceedings of the international conference on Multimedia - MM '10*, 1285–1294. https://doi.org/10.1145/1873951.1874203
14. Jörg Müller, Dieter Eberle, and Konrad Tollmar. 2014. Communiplay: a field study of a public display mediaspace. In *Proceedings of the 32nd annual ACM conference on Human factors in computing systems (CHI '14)*, 1415–1424. https://doi.org/10.1145/2556288.2557001
15. Timo Ojala, Vassilis Kostakos, Hannu Kukka, Tommi Heikkinen, Tomas Linden, Marko Jurmu, Simo Hosio, Fabio Kruger, and Daniele Zanni. 2012. Multipurpose Interactive Public Displays in the Wild: Three Years Later. *Computer* 45, 5: 42–49. https://doi.org/10.1109/MC.2012.115
16. Timo Ojala, V Valkama, H Kukka, T Heikkinen, T Lindén, M Jurmu, F Kruger, and S Hosio. 2010. UBI-hotspots : Sustainable Ecosystem Infrastructure for Real World Urban Computing Research and Business. In *Proceedings of the International Conference on Management of Emergent Digital EcoSystems (MEDES '10)*, 196–202. https://doi.org/10.1145/1936254.1936288
17. Florian Ott and Michael Koch. 2010. CommunityMirrors – Large Interactive Screens as Natural User Interfaces for Cooperation Systems. In *Workshop Proceedings of the ACM International Conference on Human Factors in Computing Systems (CHI'10): Natural User Interfaces - The Prospect and Challenge of Touch and Gestural Computing.*
18. Florian Ott and Michael Koch. 2012. Social Software Beyond the Desktop — Ambient Awareness and Ubiquitous Activity Streaming. *it - Information Technology* 54, 5: 243–252. https://doi.org/10.1524/itit.2012.0687
19. Florian Ott and Michael Koch. 2019. Exploring Interactive Information Radiators – A Longitudinal Real-World Case Study. In *Mensch und Computer 2019 Workshop Proceedings.* https://doi.org/10.18420/muc2019-ws-0565
20. Florian Ott, Michael Koch, and Alexander Richter. 2010. CommunityMirrors™ - Using Public Shared Displays to Move Information "Out of the Box." In *Competence Management for Open Innovation: Tools and IT Support to Unlock the Innovation Potential Beyond Company Boundaries*, Joachim Hafkesbrink, Ulrich H. Hoppe and Johann Schlichter (eds.). EUL Verlag, Lohmar, Köln, 141–169.
21. Peter Peltonen, Esko Kurvinen, Antti Salovaara, Giulio Jacucci, Tommi Ilmonen, John Evans, Antti Oulasvirta, and Petri Saarikko. 2008. "It's Mine, Don't Touch!": Interactions at a Large Multi-Touch Display in a City Centre. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI'2008)*: 1285–1294. https://doi.org/10.1145/1357054.1357255
22. Thorsten Prante, Carsten Röcker, Norbert Streitz, and Richard Stenzel. 2003. Hello.Wall – Beyond Ambient Displays. In *Adjunct Proceedings of the 5th International Conference on Ubiquitous Computing (UBICOMP'03)*, 277–278. https://doi.org/10.1.1.58.3459
23. Yvonne Rogers, William R Hazlewood, Paul Marshall, Nick Dalton, and Susanna Hertrich. 2010. Ambient influence: can twinkly lights lure and abstract representations trigger behavioral change? *Proceedings of the 12th ACM international conference on Ubiquitous computing*: 261–270. https://doi.org/10.1145/1864349.1864372
24. Jan Schwarzer, Lorenz Barnkow, and Kai Von Luck. 2013. Förderung der Anerkennung in agilen Softwareentwicklungsprozessen. In *Proc. Gemeinschaften in Neuen Medien (GeNeMe2013)*, 185–188.
25. Jan Schwarzer, Susanne Draheim, and Kai von Luck. 2015. Förderung von informellen Kontexten und Awareness in Scrum-Teams. In *Proc. Mensch und Computer 2015*, 13–22.
26. Jan Schwarzer, Kai von Luck, Susanne Draheim, and Michael Koch. 2019. Towards Methodological Guidance for Longitudinal Ambient Display In Situ Research. In *Proc. Europ. Conf. on Computer-Supported Cooperative Work.* https://doi.org/10.18420/ecscw2019_ep07
27. Ben Shelton and Keith Nesbitt. 2017. Evaluating WaveWatch: an ambient display of web traffic. In *Proceedings of the Australasian*

Computer Science Week Multiconference, ACSW 2017, Geelong, Australia, January 31 - February 3, 2017, 9:1--9:9. https://doi.org/10.1145/3014812.3014821

28. Dave Snowdon and Antonietta Grasso. 2002. Diffusing Information In Organizational Settings - Learning From Experience. In *Proceedings of the ACM Conference on Human Factors in Computing Systems (CHI'02)*, 331–338.

29. Mettina Veenstra, Niels Wouters, Marije Kanis, Stephan Brandenburg, Kevin te Raa, Bart Wigger, and Andrew Vande Moere. 2015. Should Public Displays be Interactive? Evaluating the Impact of Interactivity on Audience Engagement. In *Proceedings of the 4th International Symposium on Pervasive Displays - PerDis '15*, 15–21. https://doi.org/10.1145/2757710.2757727

30. Julie R. Williamson and John Williamson. 2014. Analysing Pedestrian Traffic Around Public Displays. *Proceedings of The International Symposium on Pervasive Displays - PerDis '14*: 13–18. https://doi.org/10.1145/2611009.2611022

31. Ulrich von Zadow and Raimund Dachselt. 2017. GIAnT: Visualizing Group Interaction at Large Wall Displays. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems - CHI '17*, 2639–2647. https://doi.org/10.1145/3025453.3026006