Exploring Customisation of Chained Public and Semi-Public Displays

Master Thesis

Milan Bombsch
<bombschm@student.ethz.ch>

Prof. Dr. Moira C. Norrie
Amir E. Sarabadani Tafreshi

Global Information Systems Group
Institute of Information Systems
Department of Computer Science
ETH Zürich

18th January 2017

ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

globis

Abstract

The ubiquity of public and semi-public displays has caused chained displays to become a commonplace sight. Chained displays are multiple displays next to each other forming a logical unit. Such a configuration introduces new challenges on content visualisation, as it is not clear how the content should be best spread among those screens. We investigated different ways of customising content visualisations and what effect such customisations have on the viewers. We introduce five customisation models for chained displays: *Distributed, Mirrored, Parameter, Sequential* and *Unity*. With a lab study we looked into the impact of those customisations on viewers' engagement as this is an important aspect for a successful public display deployment. The study shows that the currently widespread customisations (Distributed and Mirrored) are not the best options in terms of viewers' engagement and both Parameter or Sequential provide much better general properties. With another user study, we show that adapting the content to the viewer's walking direction is important for the sequential customisation to work. Based on the user studies we provide design guidelines for practitioners and researchers who want to deploy a chained display configuration. Those design guidelines highlight the benefits and deficits of the different customisations. We also provide an overview of the ScreenPress platform, a PDS developed in our group for rapid prototyping of PDS applications. Furthermore we show how we extended ScreenPress to support chained displays. By thoroughly separating content and visualisation through services and templates we provide a flexible architecture. Developers can either provide new content through services or design templates supporting chained displays. The ScreenPress PDS then allows users to easily manage those chained displays and assign different customisations to different services.

Contents

1 Introduction

Public displays are displays accessible and viewable by the general public. They can often be found at train stations or shopping streets and target passers-by. They provide a lot of different content like news, advertisement, travel information or weather forecast. An example of a public display can be seen in Figure 1.1. As displays get cheaper, more and more public displays pop up in all kinds of places. This changes the way people perceive those displays and also opens up new possibilities for public displays. Not only does the number of displays increase in public places, but also in semi-public places. Such semi-public displays [12] at work places, museums or schools target smaller and more uniform groups of viewers. A picture of the semi-public display of our research group can be seen in Figure 1.2. Semi-public displays typically display information relevant only to a specific group of viewers, while public displays have a much more diverse viewership. So although public and

Figure 1.1: A public display at the train station in Zürich, displaying advertisement.

Figure 1.2: A semi-public display from our group showing news about publications.

semi-public displays are similar concepts, they take different roles when it comes to viewers' privacy or viewers' interest in certain types of information.

Public or semi-public displays can be broadcast systems or interactive systems. Broadcast displays only show information like text, images or video without any control by the viewer. Interactive displays provide options for viewers to interact with the display and influence its content. Interactive displays are a highly researched topic and there are many different concepts of how interaction with public displays can be realised. There exist subtle interaction methods like sensing the viewer's distance to the display as well as more direct interaction methods like touch or gesture control. However, this thesis focuses on broadcast displays.

As the number of displays in public spaces increases, so does the number of displays per location. Displays are in a so-called chained configuration [29] if they are next to each other and form a logical unit. They do not need to be directly next to each other, but the viewer should perceive them as belonging together. There are many different possibilities to align and spread such displays along a trajectory, but the most common one is the flat configuration. In this configuration the displays are on a straight line, all pointing in the same direction. This configuration can be found quite often with varying customisations of the shown content.

A configuration is the way the displays are placed physically at the location. Their orientation relative to each other defines in what kind of chained display configuration they are. They might for example be aligned in a concave or convex half circle, flat on a straight line or slightly rotated on a straight line.

A customisation is the way the content is shown on these displays. This is how the visualisation of the content does make use of the chained displays. Examples are all displays showing the same content or spreading the content among them.

Public or semi-public displays are typically managed remotely by a pervasive display system (PDS). This enables the coordination and content management for larger networks of displays. Although today each public display provider has its own network of displays, in the future they may be connected to a greater Open Display Network [8] where multiple parties can contribute content, applications and displays. Display providers, application developers and users rely on each other for a successful engagement of the user into the viewing experience.

1.1 Problem Statement

The viewers' engagement is a key part for a successful deployment of a public or semi-public display system. If the viewers ignore the display, this is of course a waste of effort for the providers of the display. Researchers are investigating all kinds of different methods to increase the users' engagement and provide concepts for public displays which try to satisfy the display owners', the developers' and the viewers' needs. Although there is a lot of research about viewers' engagement in public displays, especially together with different interaction methods with public displays, most of them focus on single display installations.

Chained display configurations are, however, highly popular and widely deployed. A concept is missing, how content should be customised for chained configurations to maximize the viewers' engagement and satisfaction with the system. Today, most chained configurations

either display the same content on each display, or distribute multiple applications among them with one application per display. From a developer's perspective, those are the simplest ones to implement. This is, because one either puts multiple identical setups next to each other, or configures each display to show a completely different application. It is, however, not clear if those customisations are also the most effective ones with regard to conveying information and what their effects on the viewers' engagement are.

There is also no established guidance on how a PDS should support chained display configurations and what developers need to do to make their applications compatible with chained displays.

1.2 Contribution

We present customisation models for chained public and semi-public displays. These models cover different scenarios and use cases for chained displays. We provide a detailed overview of those models and their features. We identified five customisations: Distributed, Mirrored, Parameter, Sequential and Unity. Based on two user studies and a thorough analysis of their results, we provide design guidelines for practitioners as well as researchers for deciding which customisation is best for a given scenario. For each customisation we describe its benefits and deficits. We show, that the status quo, which is that most chained display configurations use either the distributed or mirrored customisation, is suboptimal and can be improved by changing some of those customisations to a parameter or a sequential customisation. We also demonstrate that adapting the logical order of the displays to the viewers' walking direction can greatly improve the usability of a chained display system.

We implemented support for chained displays and our customisation models in the ScreenPress Platform, a PDS for rapid prototyping of public display applications developed in our group. We show how such a system can support chained display configurations and how a developer can provide applications, which support different kinds of customisations.

1.3 Structure of This Document

This thesis is structured in the following way:

This chapter provides an introduction into the field of chained displays, explaining all the necessary terminology, describing the problem statement and our research contribution.

In Chapter 2, we provide a review of existing research areas relevant to this work.

Chapter 3 describes the customisation models and how we developed them.

Chapter 4 details our implementation of chained display support in the ScreenPress PDS.

Chapter 5 describes the sample applications we programmed to verify that our implementation of chained display support in the ScreenPress PDS provides developers with a basis to create applications targeted for chained display configurations.

In Chapter 6, we describe the design and results of a user study we conducted to identify benefits and deficits of the different customisations.

Chapter 7 presents a second user study we did. We tested how viewers experience chained displays, which adapt their logical ordering to the viewers' walking direction.

In Chapter 8, we present design guidelines for chained public and semi-public displays, which we developed based on our user studies.

Chapter 9 provides a conclusion of our work, summarising the steps we took and what results we achieved. Limitations and possible future work are outlined in this last chapter as well.

Appendix A provides the questions we asked during the first user study and Appendix B a tutorial on how to install ScreenPress together with our chained display extension.

2
Related Work

There is little research on chained public displays to date. Although there are many chained configurations deployed in the wild, the research community mainly focused on interaction techniques with public displays, rather than on their configurations and content visualisations. Our work therefore builds mainly on research about viewers' engagement with single public displays, design for such displays and user-centred evaluation techniques.

2.1 Viewers' Engagement With Public Displays

When designing applications or customisation strategies for public displays, one needs to understand when and how people pay attention to such display systems.

Whether people engage or not, depends on what they expect to see on the display. Public displays most often show boring advertisement, so people start to ignore them [18]. This is similar to the banner blindness effect on websites [7]. Another reason for people to ignore public displays could be an information overload, as described by Eppler et al. [9].

Huang et al. discovered that viewers often look at something interesting near a public display and only then at the display itself. People seldom look at the public display first [11]. This motivates the use of chained displays, as it potentially allows to guide a viewer's attention to the next display, once he looked at the first, therefore effectively keeping his attention. Huang et al. also observed that people spend more time when looking at smaller displays, possibly because they feel less exposed compared to when looking at a large display which many others can also observe at the same time [11]. Therefore, the use of multiple small displays might be beneficial compared to using one large display.

Florian Alt reports in his thesis [1] that it takes people some time to notice a public display. If the display is too short, they need to walk back to investigate the content. Chained displays can display the same or similar content and a passer-by can just stop and look at the display

currently next to him. Müeller et al. observed such an effect in their field study where they had multiple displays along the same trajectory [17]. Chained displays are also beneficial for passers-by who cannot stop (e.g. car drivers or cyclists). In this case the first display(s) can be used to just attract the attention of the viewer and only the following displays provide the information (e.g. traffic jam ahead, re-routeing).

With chained displays, the displays are separated by a clearly visible border of the two bezels of neighbouring displays, as well as by possible space between the displays. This could have a positive effect on viewers' engagement. This is because, Peltonen et al. mention that larger screens do not necessarily attract more people at the same time. The reason for this is that people are waiting for "their turn". On the other hand, with multiple, clearly separated displays people may not be concerned with distracting others who interact with the other displays, and therefore engage right away [25].

The distance between the viewer and the display also plays an important role for viewer's engagement. There are four different zones described by Ott et al.: *Interaction Zone, Communication Zone, Notification Zone* and *Ambient Zone* [24]. In the interaction zone, the viewer is fully interacting and engaged with the system, while viewers in the ambient zone are aware of the display, but do not intend to engage with it.

2.2 Designing for Public Displays

When designing for public displays, one needs to consider the concept of calm technology [30], as the displays have many different kinds of viewers and are not the main device of any of its viewers.

The location of the public displays is also essential to the design of the applications running on it, as lighting and contrast conditions as well as other location-based factors influence how well people are able to view and interact with the displays [24].

2.3 Chained Public Displays

Koppel at. al examined the effects of different chained display configurations [29] and found that a flat configuration triggers the strongest honeypot effect of all the tested configurations, which supports our idea of further investigating this configuration.

Grudin reports that users in a multi-monitor setup rarely expand a window across multiple displays. Instead, they keep their main task to one and the side tasks to the other displays [10]. This suggests that for multiple public displays next to each other, people will also focus on one display at a time. A similar result was reported by Bi et al. with their tiled displays setup [4]. Chained public displays therefore affect viewers differently than large public displays. Mackinlay et al. propose different user interface elements and adaptations to deal with seams in a multi-monitor setup [15]. Those could also be applied to public displays in chained configurations.

2.4 Connected Public Displays

Chained public displays need to be somehow connected to each other to work, as they need to communicate what to display when. This fits well with the idea of an Open Display Network [8], where all public displays and associated sensors are connected to a greater network and are open for applications and content from many different sources. Such applications could even be developed with an explicit focus on chained public display configurations.

When deploying a public display system and deciding on the applications to show on them, one needs to consider the relevant stakeholders: display providers, content producers and content viewers [2]. Memarovic et al. provide an in-depth analysis of a deployment of a networked public display system [16].

2.5 User Centred Evaluation

Alt et al. provide an overview of evaluation types for public displays [3]. Two of them are the lab study and the field study. Ojala et al. emphasize the difference between those two for public display research [23]. And Jurmu et al. discuss and analyse how to evaluate field studies with public displays [13].

In case of questionnaires, there are two well established set of questionnaires which focus on the users perception of a system.

The System Usability Scale (SUS), developed by John Brooke in 1986, is a 10 item questionnaire, which measures the usability of a system [6, 26, 5]. As this questionnaire gives a high-level usability rating of a system in form of a System Usability Score (a score between 0 (bad) and 100 (good)), it can be used to compare all kinds of different systems.

The User Engagement Scale (UES) by O'Brien & Toms provides a more detailed view into the perception of a system by the user [21, 22]. It divides user engagement into six dimensions: *Focused Attention, Felt Involvement, Novelty, Endurability, Aesthetic Appeal* and *Perceived Usability*. Systems can then be compared and analysed with respect to these dimensions. Although this questionnaire is rather new, it has been successfully employed to assess user engagement in many different environments [22, 19, 20, 31], including large display research [14].

Storz et al. stress how important content is in a public display setting and how much it can influence the results of a study [27].

2.6 Summary

Although it might at first be difficult to see the benefits of chained displays over single large displays, research showed that there is value in the features of chained configurations. Koppel et al. started to explore those and already confirmed the positive effect that display borders can have [29]. They even reported the standard flat configuration to perform quite well in terms of attracting viewers. We will focus on this flat configuration and explore how the content customisation influences its performance. O'Brien & Toms provide a solid foundation for

exploring the performance of such a system in terms of user engagement and its various dimensions [21].

3
Exploring Customisation Models

In order to come up with good customisation models, we first needed to explore what possibilities there are to display content with a chained display configuration. We sought inspiration from currently deployed chained display configurations and possible future use cases of chained displays. This procedure lead us to a rough idea of what models might be appropriate. To get a broader view and validate our findings, we conducted a brainstorming session within our lab. This session together with some refinements lead to our final customisation models.

3.1 Brainstorming Session

By running the brainstorming session, consisting of five people from our lab, we wanted to make sure that we do not miss any important points of the chained display customisations while also generally broaden our view on the topic. It gave us valuable insights from people currently not working with public displays and therefore speaking from the point of view of users rather than of developers. We discussed applications of public and semi-public displays currently deployed, as well as what might come in the future. Figure 3.1 shows a picture of the whiteboard after the brainstorming session.

Examples from the session are:

- Weather
  - Each display shows weather information for a different location
  - Each display shows weather forecast for a different day
- Train schedule
  - Each display shows the schedule for another platform

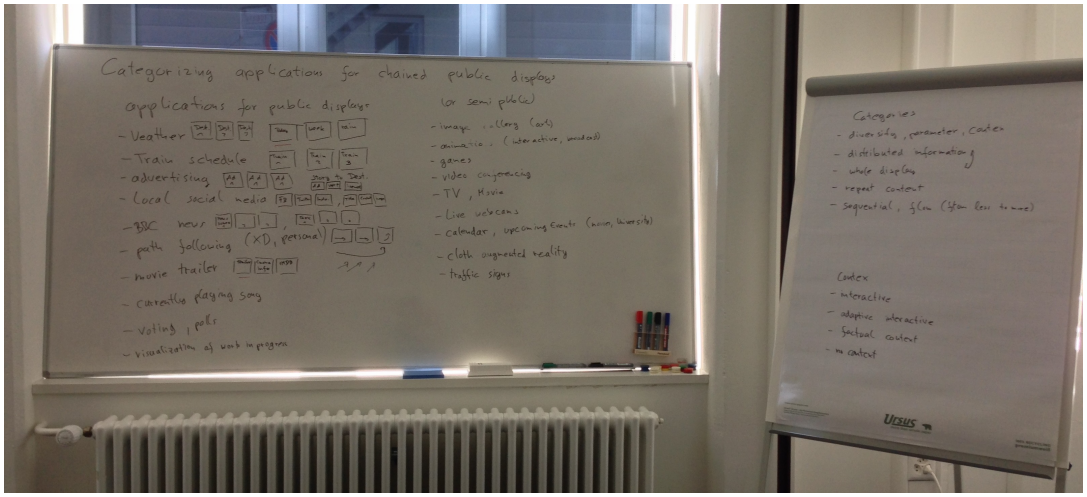


Figure 3.1: The whiteboard after the brainstorming session

  - Each display shows the schedule for another type of train (international, national, local)

- Advertisement
  - Each display shows the same advertisement
  - The displays form a storyline to the (shopping) destination
  - The advertisement is spread over all the displays

After this session we analysed their different features and grouped them by how they made use of the chained display configuration. Those groups were used to build a draft for the customisation models and were then refined by us to form the final customisation models.

3.2 Customisation Models

After the brainstorming session and our refinement, we came up with the following customisation models for chained public and semi-public display configurations:
Distributed, Mirrored, Parameter, Sequential, Unity.

The models can be partitioned into two subsets: *unrelated* and *related* customisations. Unrelated customisations display content on each display which is not related to the content on other displays. The distributed customisation is the only one belonging to this category. Related customisations are ones which display content on each display which is in some way related and coordinated to the content on the other displays. Mirrored, Parameter, Sequential and Unity are related customisations. Figure 3.2 shows a schematic representation of the five customisations, which will be described in more detail in the following subsections.

In most of our visualisations of chained displays we use four displays, as this is enough to highlight the differences between the customisations, and our lab setup also consists of four displays. The number of displays in a chained display setup is, however, not restricted to four.

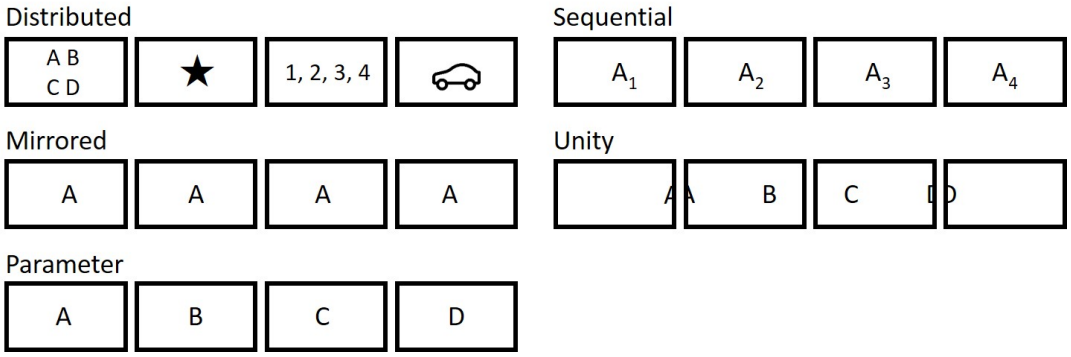


Figure 3.2: A schematic representation of the five customisations

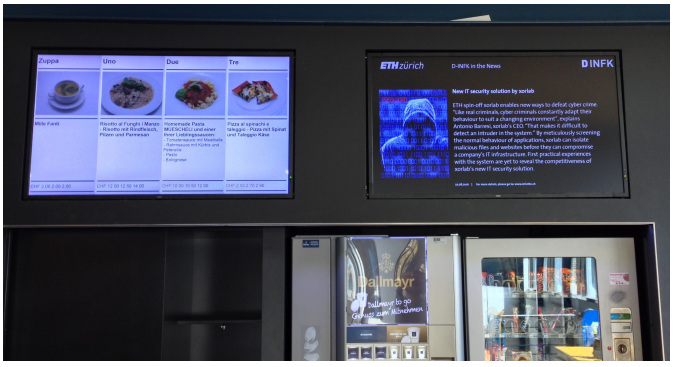


Figure 3.3: The distributed customisation used within ETHZ

Figure 3.4: The distributed customisation used within a public transport bus

3.2.1 Distributed

The distributed customisation displays unrelated content on each display. This is, every display is a separated and independent unit. The displays can show similarly styled applications, but the content must not be related. In the schematic representation in Figure 3.2 the alphabet, the star symbol, the numbers and the car image represent different, unrelated applications.

In a distributed customisation, one display could, for example, show weather information while another one displays news. If both showed some kind of weather information, this would be considered related content. The distributed customisation is the only one dealing with unrelated content, as there are no other ways for the content to not be related. The distributed customisation is among the most common ones today, as it is quite simple to put into place. There are no adjustments needed for a chained display configuration. One can simply put multiple public displays next to each other. Figures 3.3 and 3.4 show two examples of the distributed customisation. Figure 3.3 shows the cafeteria menu on one display and university news on the other one. In Figure 3.4 the left display shows news abstracts from NZZ[1] and the right one university information from ETH[2].

If there are more applications than displays, one could, for example, cycle through them

[1] `http://www.nzz.ch/`, accessed December 16, 2016
[2] `https://www.ethz.ch/`, accessed December 16, 2016

Figure 3.5: The mirrored customisation used within a shopping mall

Figure 3.6: The mirrored customisation used with four displays at a train station

with the distributed customisation. If there are fewer applications than displays, one could repeat any of those applications in any order as this does not conflict with the distributed customisation.

3.2.2 Mirrored

Another widely used customisation today, next to the distributed one, is the mirrored customisation. It simply displays the same content on each display of the chained display configuration. Figure 3.2 highlights this by putting the same content (A) on each display. The mirrored customisation displays related content.

One reason for the popularity of the mirrored customisation is probably that it is, like the distributed customisation, simple to set up. One just puts the same application on all the displays, resulting in them displaying the same content. Figure 3.5 and 3.6 show two examples of the mirrored customisation. In both cases the displays show an advertisement.

The mirrored customisation provides fewer space to content, than the other customisations, as increasing the number of displays does not increase the space for the displayed content.

3.2.3 Parameter

If the application's content is spread among several displays, but not according to any visible order, then we talk of the parameter customisation. This could be weather information for different locations on different displays or news content separated per topic. In the schematic representation (Figure 3.2) each letter stands for a different category of the content.

As mentioned before, the content should not have any particular order. People can approach the chained display configuration from any direction and still consume the content. The naming of this customisation results from the observation, that there is a changing parameter, which changes from display to display (e.g. displayed location or news topic). Two real world

Figure 3.7: The parameter customisation used within a cinema, to present different movie advertisements

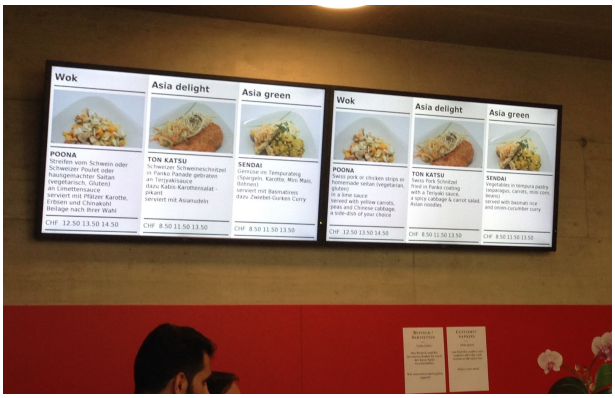


Figure 3.8: The parameter customisation used to display the cafeteria menu in different languages

examples can be seen in Figure 3.7 and 3.8. Figure 3.7 shows a picture of a chained display configuration in a cinema. Each display shows advertisement for a different movie. The parameter is the movie. In Figure 3.8 both screens display the same menu of the cafeteria, but in different languages. The left display shows the menu in German and the right one in English. The parameter is the language in which the content is shown.

The parameter customisation is sensitive to amount and type of the available content. If there are fewer categories (parameter realisations) than displays, some parameters need to be repeated. The order of this repetition is irrelevant, as each parameter is independent of the others.

3.2.4 Sequential

The sequential customisation is similar to the parameter customisation, with the exception that the content must have an order. The displays should be visited by the viewer in a specific alignment to make sense. This creates a flow for the chained display configuration and guides the viewers through the content. Most often this customisation is used to display lists of information which start on one display and continue on the next. It can also be used to provide follow up content to viewers. The first display might show the trailer of a movie and the others information about the actors. The sequential customisation can also be used to tell a part of an ongoing story on each display. Figure 3.2 visualises the sequential customisation by spreading the content of A in a sequence over multiple displays.

One example of a continuous list over multiple displays can be seen in Figure 3.9 where the second and third display show a list of departure times of commuter trains. The list on the third display continues at the position where the second display stopped.

If there is too little content to fill all the displays, with the sequential customisation, one needs to repeat the content in the same order. Otherwise the flow of the customisation would be broken.

Figure 3.9: The two displays on the right are in a sequential customisation, displaying a list of departure times.

3.2.5 Unity

The unity customisation displays related content as well. It treats the chained display configuration as if it were one wide display. Content can then be spread along this large display without caring about the borders of the individual displays. In the schematic representation (Figure 3.2) the content of A and of D overlap the border of two displays while B and C are each on one display. As the displays are treated as one large display, it may or may not happen that content overlaps the border. Figure 3.10 shows an example of the unity customisation with advertisement. In this case the displays were rotated to minimize the distance between them from the passers-by point of view.

The unity customisation can easily adapt to different amounts of content by dividing its whole view into multiple parts. Few content can be stretched with large margins, while a lot of content can be displayed by densely packing it onto the entire view.

3.3 Mixing and Nesting Customisations

Those five customisations describe all kinds of different ways to use a chained display configuration. Although they describe quite different concepts of content visualisation, they can also be mixed. For example, if we have a weather forecast for today on the first display and for tomorrow on the second display, then there is a parameter 'day' for each display, but since the days are consecutive, they form a sequence. This would, therefore, be a mixture of a parameter and a sequential customisation. Figure 3.11 shows a diagram of all the customisations and their possible mixtures. The distributed customisation is separated from the rest in this diagram, as it is the only customisation displaying unrelated content on each display.

Customisations can also be nested. Figure 3.9 shows a parameter customisation if we consider the second and third display as one unit. The first shows intercity departures and the other

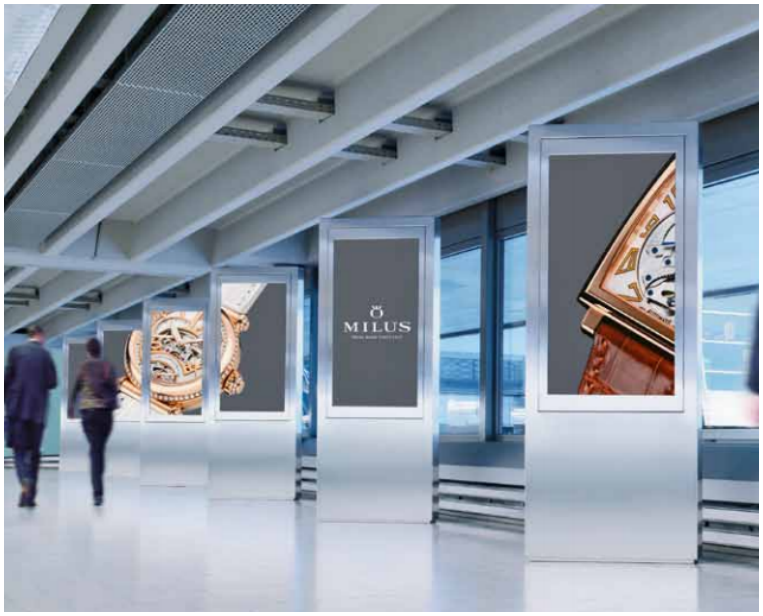


Figure 3.10: Six displays in a unity customisation at Zürich Airport (Source: `http://www.clearchannel.ch/airport-zurich-digital`)

commuter departures. However, if we look only at the second and third display, those are in a sequential customisation displaying a continuous list of commuter departures. This is, therefore, a sequential customisation nested inside a parameter customisation.

3.4 Summary

In this chapter we described how we developed our customisation models and explained them in detail. There is the distributed customisation, which displays unrelated content on each display. The mirrored, parameter, sequential and unity customisation display related content. The mirrored customisation displays the same content on each display, the parameter customisation defines the content on each display with a varying parameter, the sequential customisation imposes an order on the displays and the unity customisation uses the chained display configuration as if it was one wide display. We also laid out how these different customisations relate to each other and what their possible combinations are.

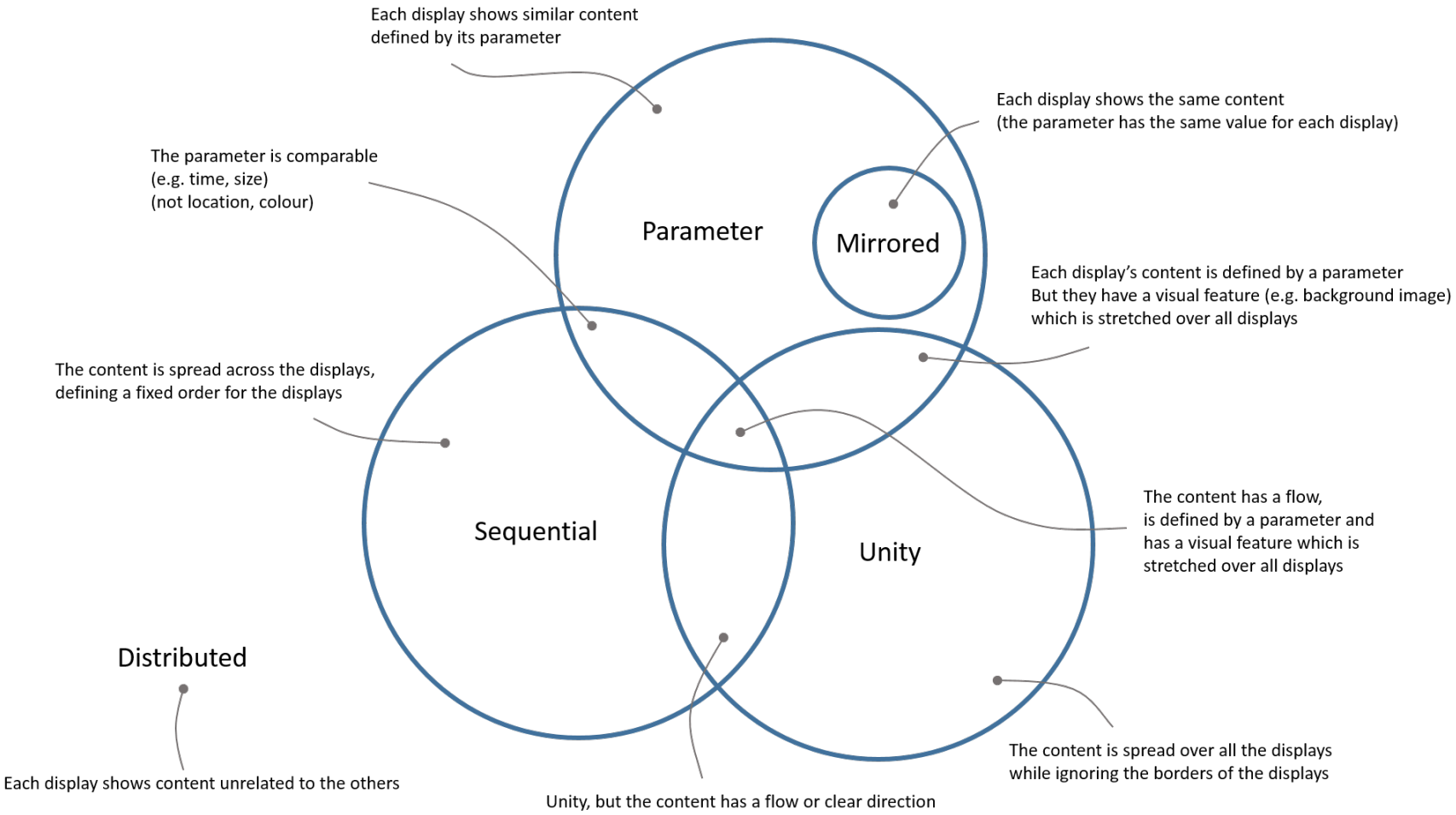


Figure 3.11: Diagram of all customisations

4 Implementation

To implement and test the customisations, we extended the ScreenPress platform, a PDS for rapid prototyping of public display applications which was developed in our group. The ScreenPress PDS had so far only been capable of handling single display configurations. Since the ScreenPress PDS is implemented using Wordpress[1], we extended it by writing a Wordpress Plugin to add support for managing chained display configurations. To give the reader a better understanding of the system, we will cover not only the parts of our extension, but briefly explain all parts of the ScreenPress platform. In the process of extending ScreenPress we also refined most of the existing code.

Since the ScreenPress PDS is already built on Web technologies, we continued this path with our extension. Taivan et al. provide a good analysis of the benefits and deficits of web technologies in the field of public displays[28]. Deficits such as the missing ability to scroll on public displays or possibly long loading times for external content need to be considered when implementing such a system. Benefits are the widespread adaptation of Web technologies in different fields as well as their ease of portability to all kind of different systems.

4.1 Meta Model

First, we explain the basics of the meta model underlying the ScreenPress PDS and then explain where and how we extended it.

4.1.1 ScreenPress PDS

Figure 4.1 shows the meta model of the ScreenPress PDS.

[1] `https://wordpress.org/`

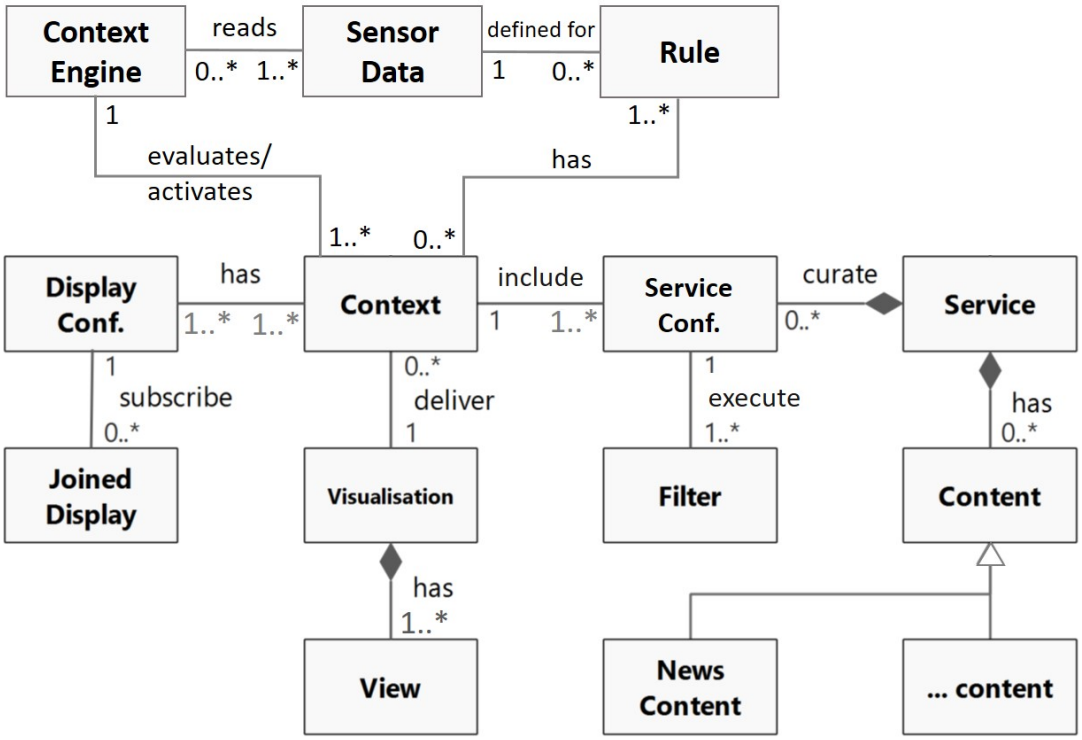


Figure 4.1: The meta model of the ScreenPress PDS.

It consists of three main parts.: The Joined Displays, the Context and the Services. The Joined Displays are the actual displays, which display the content of a specific URL. The Context decides which URL a display calls and what the content and its visualisation will be, based on a set of context defining rules (e.g. time smaller 18h, number of people in front of the displays equals 3, the temperature is high, etc.). The Services provide the actual content. They fetch it either from a database, a web resource or generate it themselves.

Which context is active is decided by the context engine. The context engine checks for each context if its rules are satisfied by the current sensor data. An example for sensor data is time and a corresponding rule could be between eight and twelve o'clock in the morning. Sensor data can come from hardware or software sensors. The active context then decides what will be shown on the joined displays.

For the active context, the service configurations curate the services using filter objects. The filtered content is then, together with a visualisation, passed to the display configurations. A display configuration is a database object to manage a display. It links the display to different context objects. Multiple actual displays can subscribe to such a display configuration. The result is, that all those displays show the same content and visualisation. A visualisation is the HTML and CSS which define the visuals of the website shown on a display. One visualisation has multiple views. Those correspond to different realisations of the same visualisation which is implemented by using responsive web design.

The ScreenPress PDS was designed with the goal of extensibility in mind. Figure 4.2 shows the meta model for the plugin integration into the ScreenPress platform. A plugin can either

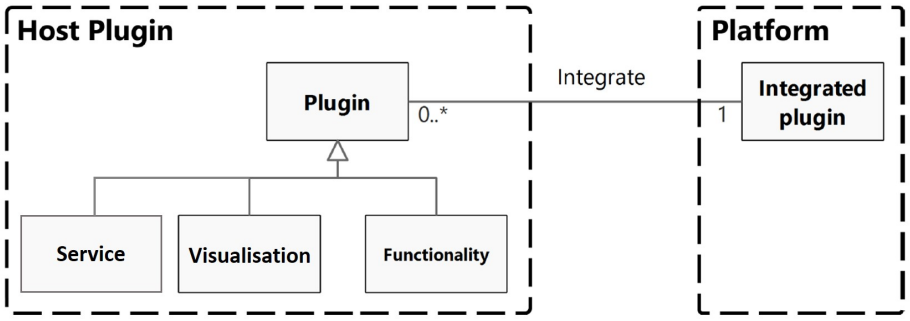


Figure 4.2: The meta model for plugin integration in the ScreenPress PDS.

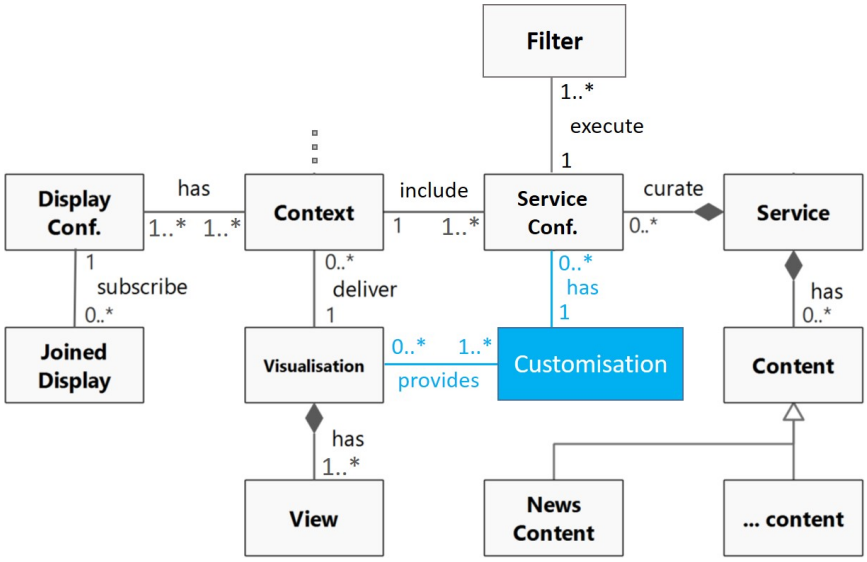


Figure 4.3: The meta model of the ScreenPress PDS with the chained display extension.

be a service, which provides content, as described above, or a visualisation, which defines how the content will be visualised on the display, or a functionality, which changes or extends the features of ScreenPress.

4.1.2 Chained Display Extension

The Chained Display Extension for the ScreenPress PDS is a functionality plugin. It enables the platform to manage chained display configurations. This is, adding and removing displays from a chained display configuration, defining their order (from left to right) and defining what content should be displayed in which customisation for a given context.

To provide support for chained displays and different customisation models, we changed the ScreenPress meta model and introduced a new class. Figure 4.3 shows the new meta model. The three grey dots above the context class visualise that the context engine part is still there (cf. Figure 4.1) but omitted in the diagram to save space.

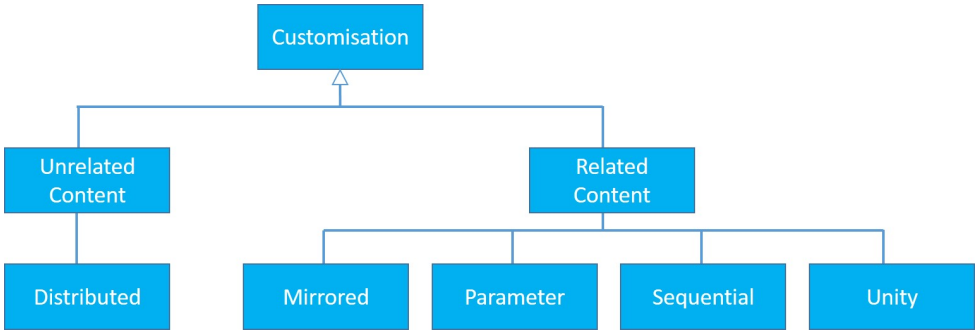


Figure 4.4: The meta model of the customisations

The new customisation class is connected to the visualisation and the service configurations. Each visualisation needs to provide one or more customisations. Each service in a given context can then be connected to one of those customisations, defining how this service will be shown on the display. A visualisation can either provide only a subset of the customisations introduced in this thesis or even introduce new ones, which will then be automatically available to all services. This enables a clear separation of content and visualisation and leads to a flexible architecture.

Figure 4.4 provides an overview of the five customisations and their aforementioned grouping.

4.2 The PDS

The ScreenPress implementation consists of three parts: The Back Server, the Front Server and the Client Side. The Back Server consists of a Wordpress installation to manage the displays, contexts, services and templates. It also stores and provides the PHP files of each service, which are responsible for fetching the content. The template files are stored on the Back Server as well. Those provide the visualisation at the client side. The Front Server forwards the data from the Back Server to the clients and also implements timers to fetch new data from the services. The client runs a template, which is able to visualise the data from each service with possible different customisations.

Figure 4.5 shows a diagram of the ScreenPress architecture together with our chained display extension. The grey part is the ScreenPress platform, which can be extended with services and templates the system can use. Services, which are coloured in red, can be put into the socket on the Back Server and will then be available for all parts of the system. A user can manage the services on the Back Server CMS, the Front Server will implement timers for each service to pull new data and the client can receive this data for visualisation. The templates (in green) are also stored and managed on the Back Server. The Front Server forwards them to the clients, which then display them and render the services' data with them. The client module part, which is coloured blue, enables the displays to communicate with the Front Server as well as with other displays in the same chained display configuration.

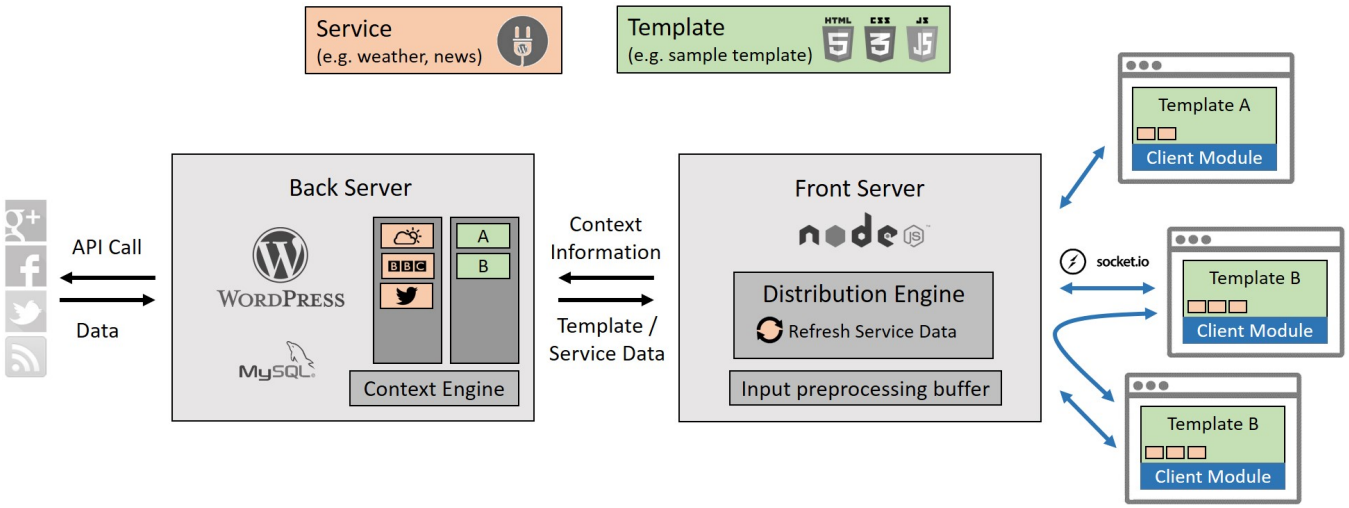


Figure 4.5: The ScreenPress architecture together with the Chained Display Extension.

4.2.1 Back Server

The Back Server manages all the display configurations, context settings, services and templates. This is done with the help of a Wordpress instance with a MySQL database[2]. Wordpress with its rich UI enables users of the system to easily manage and configure their public display settings. The Back Server also provides an API for the Front Server to retrieve the data stored in the Wordpress instance through a PHP file. The Front Server can, for example, ask for the appropriate template for a client (by calling `http://yourdomain.com/ambimajson/?start=yes&ptype=template&displayName=myDisplay`) or, if the services activated for a client have changed (by calling `http://yourdomain.com/ambimajson/?start=yes&ptype=services&displayName=myDisplay`). The Back Server's API also provides access to the individual services' code to fetch new content. For all this to work, the service and template files need to be put into the appropriate folders on the Back Server. In that case, they will be available to the rest of the system.

The context engine on the Back Server gets data about the current context (e.g. time) from the Front Server and then returns the data associated with the context object which is activated by this data (e.g. display the Sample Template in the afternoon).

The chained display extension uses the display configurations connected to a context object to define a chained display configuration. The order in which they are connected can easily be changed in Wordpress and defines their logical sequence. This sequence is then, in form of IDs, passed to templates on the clients, which can then render the content appropriately. The Wordpress UI was also extended with the option to define an individual customisation for each service in a context object.

[2] `https://www.mysql.com/`, accessed on December 13, 2016

4.2.2 Front Server

The Front Server is the glue between the Back Server and the client. We use a nodeJS server[3] for this task. It has an active connection to the client (we are using socket.io[4]) and regularly checks if the Back Server has new data for the clients. When a new client connects to the Front Server, it calls the Back Server's PHP API and asks for all the relevant information for the client to be set up. This includes the template for this client and its activated services. For each of the active services, the Front Server sets up a timer, which regularly checks for new data from this service through the Back Server. For new clients it also stores the size of their browser window to report this value to the other displays in this chained configuration. This information is needed by the unity customisation.

The input preprocessing part of the Front Server preprocesses context defining sensor values and propagates them to the Back Server. Such sensor values could, for example, be time, number of people in front of a display or outside temperature. To avoid sending the new time every second, the Front Server only sends the new time every ten seconds. Other input values can be preprocessed in a similar way or just sent to the Back Server each time their value changes. The Back Server then returns the information associated with the context object which is active for these sensor values. If the returned data for example includes a template change for a client, the Front Server informs this client and the client module loads the new template.

The Front Server, together with the client module, provides a broadcasting mechanism to clients. With this, displays in the same chained display configuration can send each other messages. This is especially useful for the sequential customisation, if the displays, for example, want to continue a list at the point the previous display left off. The way this works is that clients send `('broadcast', 'message', data)` to the Front Server which then forwards `('message', data)` to all clients in the same chained display configuration.

4.2.3 Client

The client loads a default template upon first connecting by browsing the URL `www.yourdomain.com:5000/?display=myDisplay`. It will be redirected to the correct template as soon as the Front Server has received the set up data from the Back Server.

Once the client has loaded the correct template, this template defines all behaviour of the client. It also provides the HTML and CSS code to render the content of the activated services. The client has an active connection to the Front Server through which it receives regular updates of the data of its services as well as information about the other displays in its chained display configuration.

The client module sets up the connection to the Front Server. It enables the broadcasting mechanism within chained display configurations and replaces the current template when the Front Server informs it about a template change. The client module also triggers a reload on the client side if the Front Server was restarted or the client lost its connection to it for some other reason.

[3] `https://nodejs.org/en/`, accessed on December 13, 2016
[4] `http://socket.io/`, accessed on December 13, 2016

4.3 Services and Templates

We clearly separated content and visualisation by having services and templates in order to provide a flexible architecture. A service provides content from all kind of different sources, but does not influence the visualisation of this content. This is solely the template's responsibility. A template is an HTML document which uses JavaScript to parse the content from the services and render it on the public displays. By changing the template one can completely modify the visuals of the display setup while the viewers are still able to consume all the content. At the same time, new services also work directly with the existing templates. All this is achieved by defining a structure for the data the services provide.

4.3.1 Services

A service retrieves data from any kind of data source, puts it into a specific format and provides it to templates on the client side. We implemented such services as Wordpress plugins. Thus, the services can easily be activated and deactivated for a specific ScreenPress installation. Their code is completely contained in their plugin folder and to install the service on another ScreenPress installation, one simply has to copy and paste the plugin folder of the service.

For the rest of this section we will use `My Service` as an example name for a service. This should be replaced with the actual name of a service in all instances.

Code Snippet 4.1 provides an overview of the file structure of a service. `spser` stands for ScreenPress Service. The plugin folder needs to stick to the following naming conventions: `spser-my-service`. There have to be two files in this folder: `spser-my-service.php` and `spser-my-service-data.php`

```
spser-my-service/
    spser-my-service.php
        - Plugin information // description, author, etc.
        // If necessary for the service:
            - create custom post-type
            - connect post-type to collection post-type
    spser-my-service-data.php
        - namespace my_service
        - timeInterval() -> Integer
        - getServiceData($collectionId) -> JSON
```

Code Snippet 4.1: Folder and file structure of a service

`spser-my-service.php` is the Wordpress plugin file. It should contain the necessary plugin description at the top. If the service needs to be configured or managed by the user through the Wordpress UI, a service can use this file to define functions to register a new post-type and connect it to the `collection` post-type (which represents the context objects)

using the Post 2 Post plugin[5], or create a custom configuration UI for this service.

The `spser-my-service-data.php` contains the actual content fetching part of the service. It should provide a `timeInterval()` function, which returns an integer, indicating at which rate (seconds) the Front Server should call the data fetching function. It should also provide a `getServiceData($collectionId)` function. This function has as an input the current context object (`$collectionId`) and should return the fetched data from the service in a specific format. The format is described in Code Snippet 4.2. Those two function should be wrapped in a PHP namespace of the form `my_service`.

```
// one service
[
 // one entry per parameter
 {
  "parameter": "Some parameter string",
  "data": [
   // one entry per content object
   {
    "sequenceNumber": some integer number,
    "data": [
     {
      "type": "text"|"image"|"video",
      "value": "some string"|"someimg.png"|"somevid.mp4"
     },
     ...
    ]
   },
   ...
  ]
 },
 ...
]
```

Code Snippet 4.2: JSON structure in which each service should provide its data.

[5] `https://wordpress.org/plugins/posts-to-posts/`, accessed on December 13, 2016

```
[
  {
    "parameter": "Barcelona",
    "data": [
      {
        "sequenceNumber": 1483106272,
        "data": [
          {
            "type": "text",
            "value": "Barcelona (30.12.2016)"
          },
          {
            "type": "image",
            "value": "http://aberfeldy.inf.ethz.ch/wp-content
               /plugins/spser-weather/Weather-SVG/sw-13.svg"
          },
          {
            "type": "text",
            "value": "Scattered Showers"
          }
        ]
      },
      {
        "sequenceNumber": 1483192672,
        "data": [
          {
            "type": "text",
            "value": "Barcelona (31.12.2016)"
          },
          {
            "type": "image",
            "value": "http://aberfeldy.inf.ethz.ch/wp-content
               /plugins/spser-weather/Weather-SVG/sw-01.svg"
          },
          {
            "type": "text",
            "value": "Sunny"
          }
        ]
      }
    ]
  }
]
```


Code Snippet 4.3: Example of JSON data provided by the weather service

The JSON format of the data from a service (Code Snippet 4.2) is structured in the following way: The most outer structure is an array of parameter elements (e.g. weather locations). These elements each correspond to one parameter (typically the filter objects). Every parameter element has multiple content elements (e.g. different weather forecasts) in its data array. Each such content element has a sequenceNumber which defines a sequence over all the content elements. This enables the template to correctly display the content in a sequential customisation. And at the most inner nesting step are the individual data elements (e.g. title and weather image) of a content element. One example of such a data element array would be a title, a feature image and a description text. Code Snippet 4.3 shows example data from the weather service. The sequenceNumber is in this case a timestamp corresponding to the day of the weather forecast.

4.3.2 Templates

A template is responsible for the visual style of the displays. It is an HTML document which uses HTML, CSS and JavaScript to parse and render the data it gets from the activated services. Each public display loads an URL from the Front Server which serves the activated template to the display. The display does not load anything else. The template contains all the necessary code to render the content from the services. The template includes the client module. The client module is a JavaScript file, that sets up the connection to the Front Server and handles template changes. Through the active connection to the Front Server the template receives all data updates for the services as well as other updates such as template changes or changes to the chained display configuration.

```
<customisations>
  <unrelated>Distributed</unrelated>
  <related>Mirrored</related>
  <related>Parameter</related>
  <related>Sequential</related>
  <related>Unity</related>
</customisations>
```

Code Snippet 4.4: Example of a customisation definition at the top of a template file.

At the top of the HTML file each template has to define which customisations it provides. See Code Snippet 4.4 for an example. The Wordpress UI will then allow the user to choose for each activated service one of the customisations the template provides. This provides template developers with the flexibility to either only concentrate on a subset of the customisations or even come up with new ones, which will then be automatically available to all services.

To be able to connect to the Front Server without knowing its URL beforehand, a template needs to put the string `<!--PLACEHOLDER: server url-->` at places where the URL of the Front server should be inserted. This is needed to establish a web socket connection as well as loading additional files such as CSS files. The Front Server rewrites each template

file before it gets served, replacing those strings with its own URL.

4.4 Summary

In this chapter we described the architecture and implementation of the ScreenPress PDS and our extension. Thanks to its flexibility, we were easily able to extend ScreenPress to support chained display configurations. A user can now configure such chained displays in the Wordpress UI and set the active services as well as define in which customisation they should be shown on the displays. By choosing a template the user can decide on the visual look of the displayed content.

5 Sample Applications

In the following sections we will list and describe the sample applications we developed using ScreenPress and our chained display extension. First we developed the Lucky Numbers service and template to ensure all the features needed for chained display applications exist and we are not missing something in our chained display extension for ScreenPress. The Group Articles service in combination with another template were developed to be able to conduct a user study and evaluate the effects of the different customisations on viewers. Special care was taken to have static content and to always have the same content in each customisation for the user study. The Weather service, BBC News service and the Sample Template were developed to further prove the flexibility of our system and its possible use in an actual public display deployment.

5.1 Prototype: Lucky Numbers

The Lucky Numbers service and its corresponding template were kept as simple as possible. They were developed to ensure that all the system components work as intended and are flexible enough to enable the management and displaying of different customisations in a chained display configuration.

The Lucky Numbers service lets users enter their lucky numbers through ScreenPress. The custom post-type for this service only has a title which contains the lucky number. The Lucky Numbers template then displays those numbers. An overview of all the customisations with the Lucky Numbers template can be seen in Figure 5.1.

The distributed customisation shows existing ScreenPress services with their own template since this was our first service with the new architecture.

The mirrored customisation shows the first lucky number connected to the current context object on all the displays.

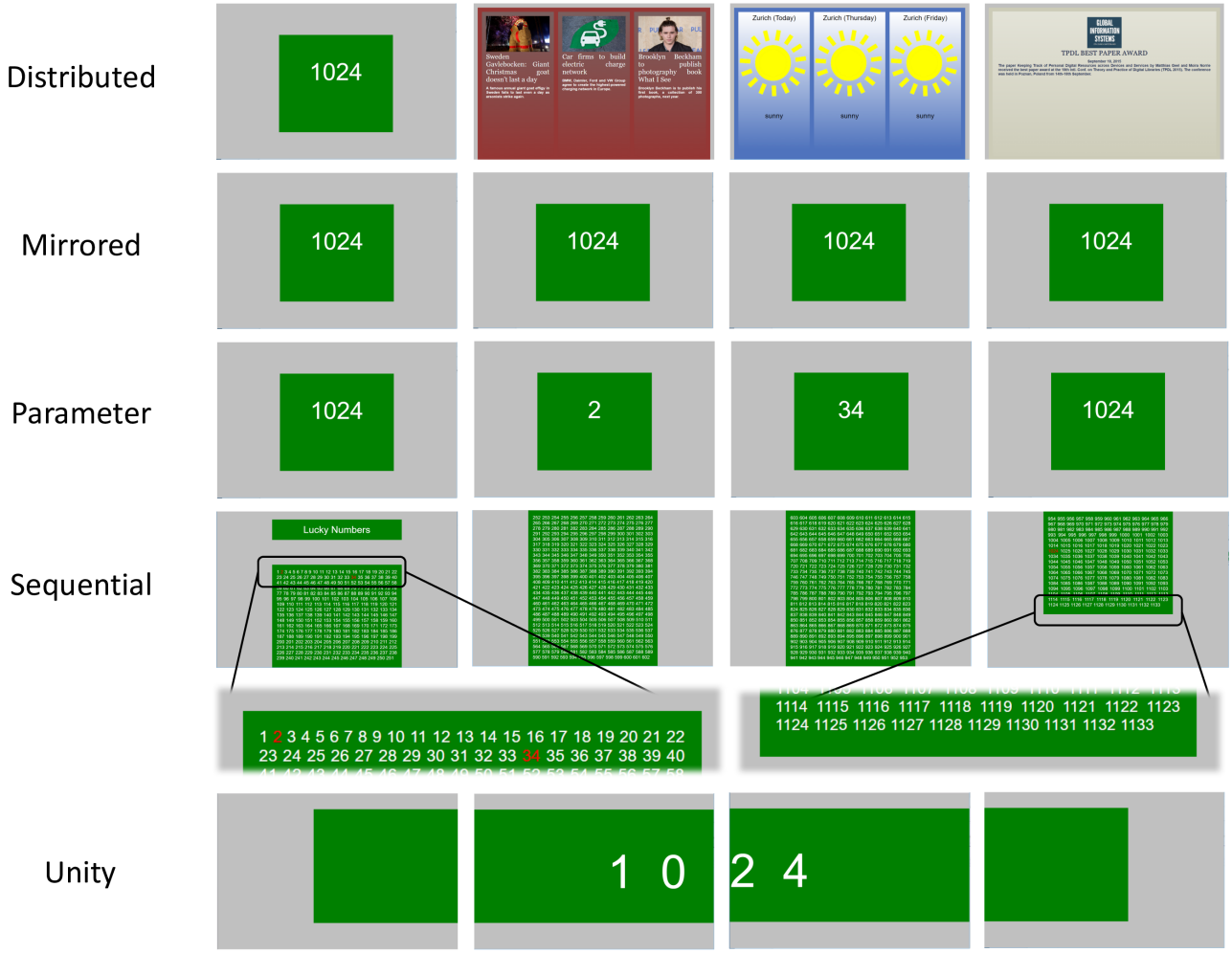


Figure 5.1: An overview of the Lucky Numbers service and template in all customisations.

The parameter customisation shows different lucky numbers on each display and starts from the beginning again if there are more displays than lucky numbers.

With the sequential customisation we wanted to test if we could have an ongoing list over multiple displays. We did this by displaying a sequence of numbers from 1 to as far as we get with the available screen space and each time we encounter an activated lucky number we display it in red. This continuous list is achieved by using the broadcast function of the Front Server to communicate how far the list got on the previous display, which depends on the size and proportions of the display.

The unity customisation takes the first connected lucky number and scatters it over all the displays. To achieve this we render the whole view on all the displays and then zoom in to the appropriate position.

Since the service was able to store and retrieve the lucky numbers and the template could display them in all the five customisations, we can be positive that our chained display extension to the ScreenPress platform is powerful enough to enable developers to target chained display configurations.

5.2 Group Articles

The Group Articles service allows users of ScreenPress to show articles to the group of viewers of the public displays. The corresponding template displays those articles in a nice layout.

We developed this service and template with our user study in mind. The service is therefore static and does not fetch any external, changing content, but rather uses the articles typed into ScreenPress by the user. This ensures every participant of the user study would see the same content. The template shows exactly the same content with each customisation to avoid a bias in the results due to different content.

Figure 5.2 provides an overview of all the customisations with the Group Articles service and template.

The distributed and mirrored customisation show all four parts of the article on one screen.

The parameter customisation puts each independent part of the article on a separate display, which results in one image and one paragraph on each display.

The sequential customisation aligns the four parts of the article next to each other while also separating images and text. This gives the content a flow from left to right.

The unity customisation puts the title of the article in the middle and aligns the text pieces and images around it, resulting in multiple text fragments ranging over multiple displays.

For this figure we used the giraffe article which was one of five articles used in the user study. The user study is described in Chapter 6.

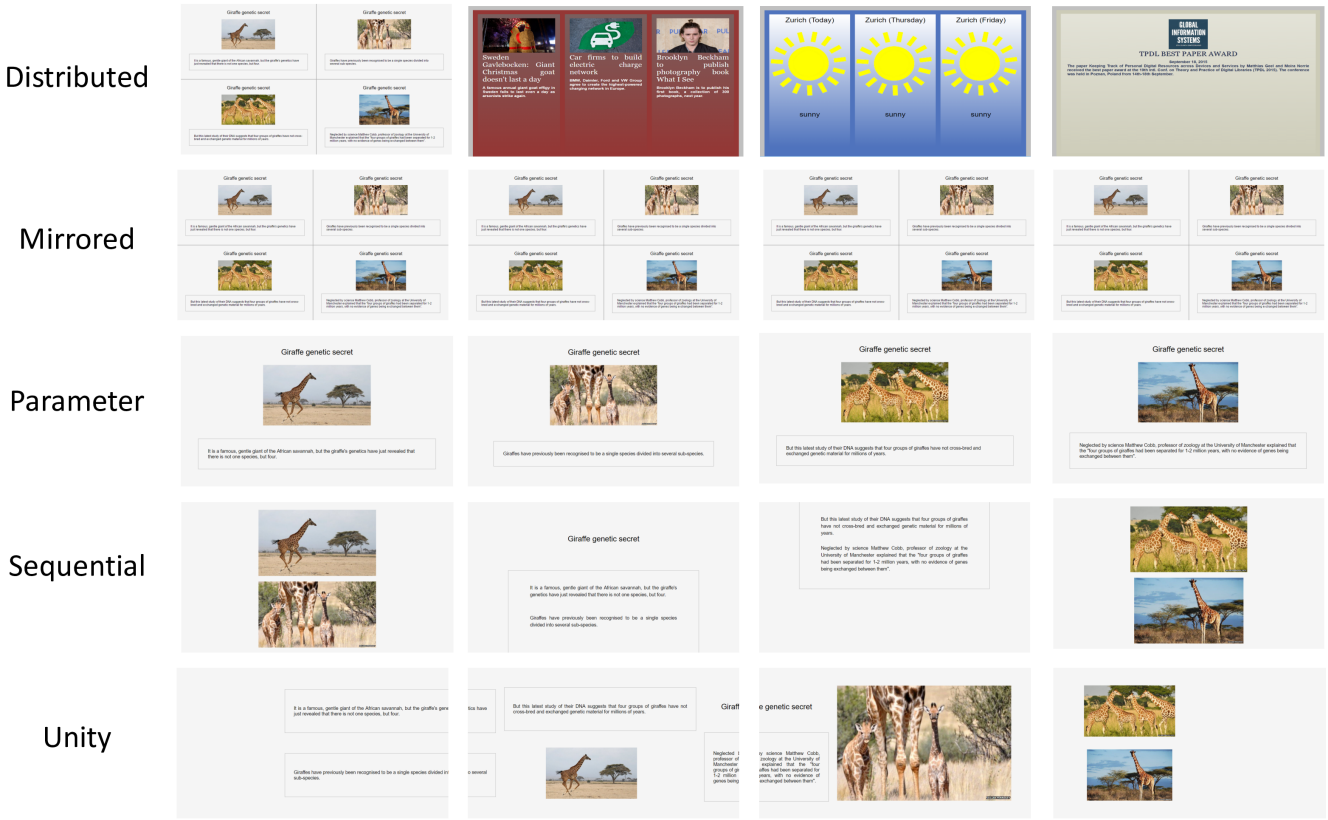


Figure 5.2: An overview of the Group Articles service and template in all customisations.

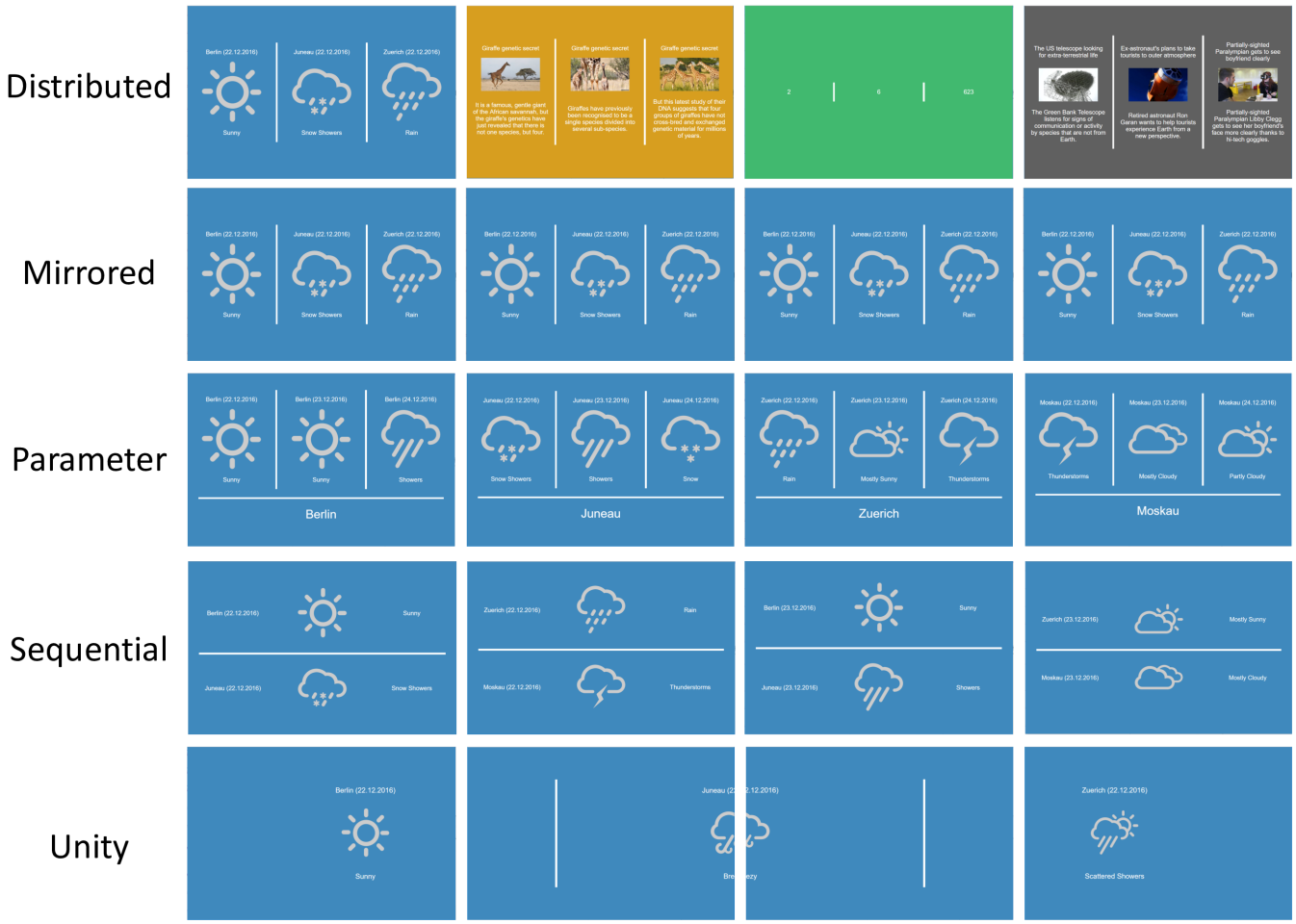


Figure 5.3: An overview of the Weather service shown with the Sample Template in all customisations.

5.3 Weather Service

The Weather service fetches weather data from the Yahoo Weather API[1] for locations the user entered into ScreenPress. The custom post-type has, in this case, only a title, which contains the name of the city the weather data should be displayed for.

The service uses the weather data to get the forecast for the next three days including today. It then extracts from this data the forecast text (”Sunny”, ”Cloudy”, etc.) and attaches a suitable image from the set of weather images from Daniel Vierich[2] to the data. One data element array consists of the location together with the date in brackets, the weather image and the forecast text. Figure 5.3 shows the Weather service with the Sample Template in all the customisations.

[1] `https://developer.yahoo.com/weather/`, accessed December 15, 2016
[2] `http://www.danvierich.de/weather/`, accessed on December 15, 2016

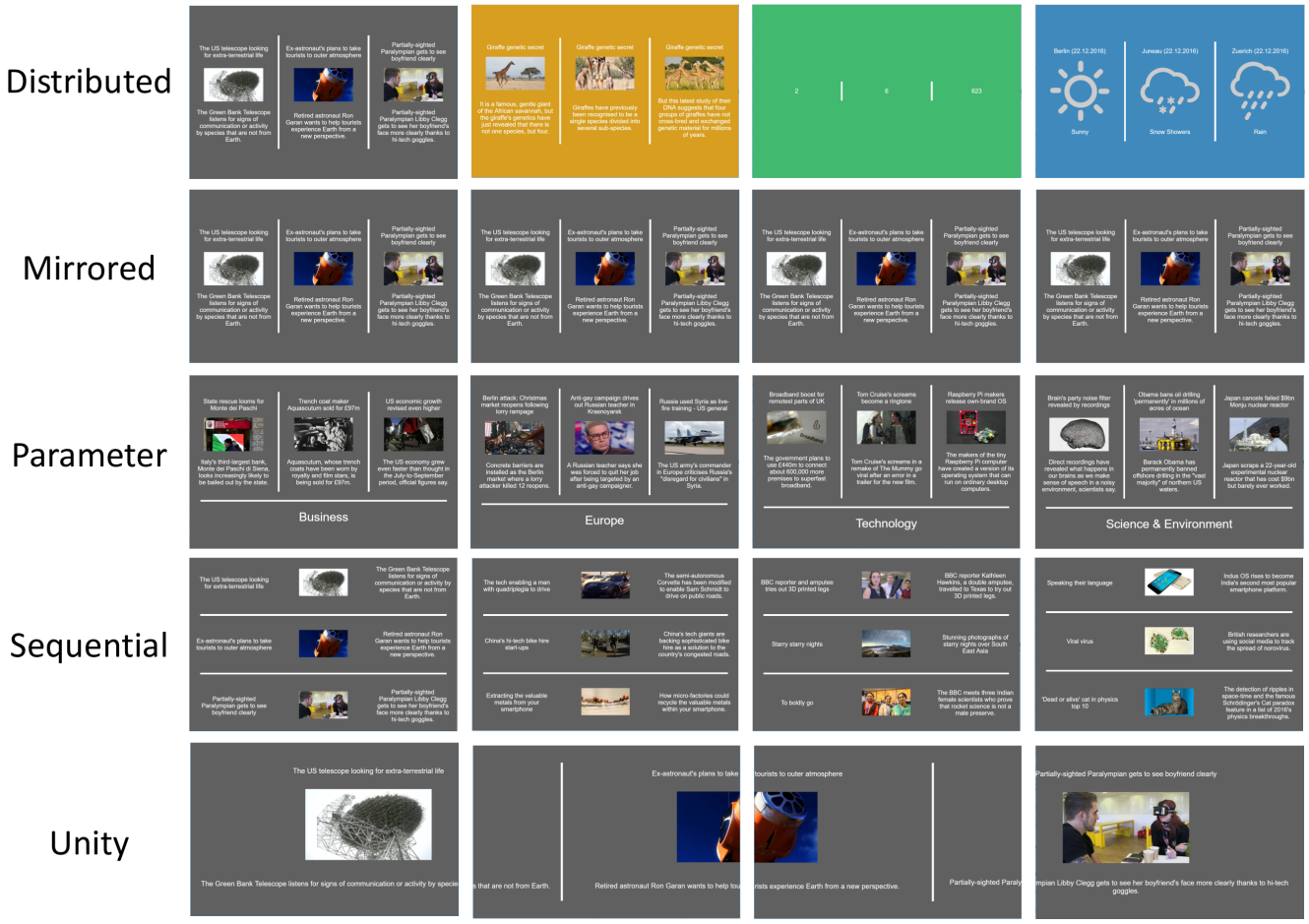


Figure 5.4: An overview of the BBC News service shown with the Sample Template in all customisations.

5.4 BBC News Service

BBC News is a service which integrates news headlines from BBC News[3] into the ScreenPress PDS. Users can type in RSS feed URLs into the custom post-type and the service will fetch and parse the news from this feed and provide it in the appropriate JSON format. For this custom post-type we programmed a custom meta box (a Wordpress UI element to edit a post) for the user to enter the URL of the news feed. This demonstrates that ScreenPress can easily be used to provide the users of the system with custom input fields depending on what the service needs to retrieve its data.

The service parses the news feeds and extracts the headline, the thumbnail and a short description of the news article. Those are then in this order packed into the service JSON format and forwarded to the template.

Figure 5.4 shows the BBC News service with the Sample Template in all customisations.

[3] `http://www.bbc.com/news`, accessed December 15, 2016

5.5 The Guardian Service

We programmed another news service, which uses The Guardian[4] as source. This service extracts whole articles from The Guardian website and provides them together with a thumbnail. This service is useful if you have large displays, which can display a lot of text, or if the template spreads the text over multiple displays. The One News Template (described in Section 5.7) does, for example, spread the whole content of a single article across all displays in the sequential customisation.

5.6 Sample Template

With the Sample Template we wanted to develop a flexible template. It is able to display any number of data entries inside a content part from a service as long as they fit on the screen. The basic layout is a three column structure. Each column gets filled with one content bit. It takes data element after data element and aligns them from top to bottom in the order they are stored in the JSON array. In case of the BBC News service this is: headline, thumbnail, description. For the Lucky Numbers service only one lucky number will be displayed per column. But if another service would have ten images per content bit, they will be aligned until the column reaches the bottom of the screen. Figures 5.3 and 5.4 show the Sample Template with the Weather and the BBC News service in all customisations.

For the distributed customisation each service will be displayed in a three column layout showing the three content bits with the highest sequence numbers for each service. Each service will be displayed with a different background colour, which is chosen from a fixed colour array. In the examples (Figures 5.3 and 5.4) the distributed customisation shows the BBC News, the Weather, the Lucky Numbers and the Group Articles services.

For the mirrored customisation each display, of course, shows the same content. In this case it displays the same three column layout as for the distributed customisation, except it only displays one service at a time.

The parameter customisation displays the content for each parameter on the displays along with the name of the parameter at the bottom.

The sequential customisation aligns the content bits in a long list ranging over all the displays. If one content bit does not fit entirely on the screen, it will be shown on the next display. Thus, if we had a display in portrait mode we would see more than 2 content bits per display.

The unity customisation uses the same column layout as the mirrored customisation, but spread over all the displays. The number of columns in this case, is the number of displays minus one.

If one chained display configuration is connected to multiple services and each should be displayed in a related customisation, the Sample Template cycles through them.

[4] `https://www.theguardian.com/`, accessed January 12, 2017

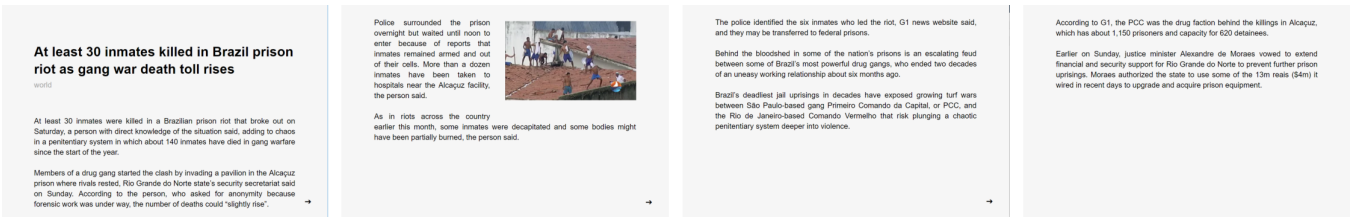


Figure 5.5: An overview of The Guardian service shown with the One News Template in the sequential customisation.

5.7 One News Template

The One News Template is a template to display a single news article. This template only supports the sequential customisation. The article is spread across all the displays in the chained display configuration. The first display shows the headline, the category and the beginning of the article. The article continues on the other displays and, if possible, is equally distributed among them. The second display in the configuration additionally shows the thumbnail of the article. If a service provides multiple articles, this template will cycle through them, displaying one at a time.

5.8 Summary

This chapter shows that our extension to the ScreenPress PDS is able to provide developers with a powerful basis to develop chained display applications consisting of different combinations of services and templates.

The Lucky Numbers and the Group Articles services and templates show that a large range of different visualisations are possible. The BBC News service, the Weather service and the The Guardian service provide examples for integrating external content which then can be visualised in different ways. The Sample Template shows, how a flexible template is able to deal with a lot of different kinds of content.

We look forward to see new services for the ScreenPress platform and interesting visualisations through new templates by developers and designers.

6 User Study

To explore the effects the different customisations have on their viewers, we conducted a user study. In this section we will describe the study design, execution and analysis. As this user study was a key part of the thesis we spend a lot of time carefully preparing and implementing the different parts. First we look at the study design and the reasoning behind it. The results section contains the results of the study and our statistical analysis as well as a summary of the qualitative feedback we collected. We discuss those results in the discussion section, which is structured similar to the results section. At the end of this chapter we provide a summary of the user study.

6.1 Study Design

The goal of this user study was to explore the benefits and deficits of different customisations with respect to viewers' engagement. We decided to do a lab study instead of a field study in order to have better control over possible influencing factors. With a lab study we could also focus our attention on one participant at a time and get a more detailed picture about his engagement and the differences between the customisations.

6.1.1 Setting

The study was carried out in our lab's meeting room. Our chained display configuration consisted of four displays aligned next to each other. They were mounted on shelves so as to be approximately at eye level for people standing in front of them. Figure 6.1 shows a picture of the chained display setup with a participant in front of it.

The displays where each connected to a separate computer which were running the *Windows*

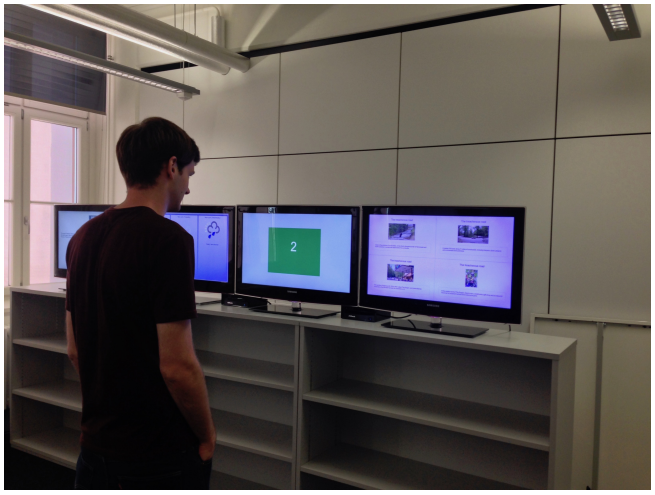


Figure 6.1: A participant looking at the chained display configuration of the user study

Figure 6.2: A picture of the computer participants used to fill out the questionnaires.

7[1] operating system. They were configured to show the respective ScreenPress URLs with the *Chrome Browser*[2] in kiosk mode. This mode hides all interface elements and solely displays the website.

Participants had to fill out multiple questionnaires. This was done at a computer in the same room as the chained display setup. A picture of the place participants filled out the questionnaires can be seen in Figure 6.2.

The Group Articles app (described in Section 5.2) was used for the study. The distributed customisation also showed the 'Group News' and the 'Weather' service from the ScreenPress platform as well as the Lucky Numbers app (described in Section 5.1).

[1] `https://en.wikipedia.org/wiki/Windows_7`
[2] `https://www.google.ch/chrome/browser/desktop/`

6.1.2 Task

Each participant had to fill out a pre-study questionnaire, then complete a reading task with each customisation and fill out two questionnaires after each customisation (one about the article and one about the customisation) and at the end fill out a post-study questionnaire. We estimated a duration of 45 minutes per participant to complete the user study. During each reading task, participants were allowed to freely walk around and complete the task without any time limit for looking at the displays. While participants answered the questions on the computer the chained displays showed a white background, so that participants had to answer the questions using their memory of the displays. With this we could test if they remembered the content on the displays.

Before they started with the pre-study questionnaire, we gave them a short introduction on what public and semi-public displays are, what a chained configuration and a customisation is. We also told them the process of the study. We did not introduce them to the individual customisations, as we did not want to influence them on their attitude towards any of the customisations.

For each of the customisations the participants had to read a short article of 4 to 8 sentences. Each article also included four pictures. The choice of a reading task ensured that the participants spend enough time with the displays to get to know the individual customisations. We had five articles: Giraffe[3], Sabbatical[4], Wolves[5], Cricket[6] and Motorcycle[7]. All of them were short versions of articles from `bbc.com`, a British news website. For each participant the pairing of article and customisation was randomised to ensure that the result for a customisation is not influenced by the quality of a particular article.

Every participant was presented with the distributed customisation first. This happened for two reasons: First the distributed customisation is widely used in practise and people are therefore familiar with it. This provides a good starting point to get them used to the system. Second, as the distributed customisation displays unrelated content, it is the only one which displays more than only the article. It also displayed weather information for New York, a news article about a publication from our group, and the Lucky Numbers app displaying the number 2. If people now encountered this customisation in the middle of the study, they might only concentrate on the article as this is the same as in the other customisations, and therefore introduce a bias in the results. The bias of always being the first one is as said weakened by the fact that the distributed customisation is familiar to the participants. The results from the post-study questionnaire (especially Figure 6.8) emphasises this fact.

The other four customisations were presented to the participants in random order after they had seen the distributed customisation to avoid a bias introduced by the order in which participants learn about the customisations.

[3] `http://www.bbc.com/news/science-environment-37311716`
[4] `http://www.bbc.com/capital/story/20160325-the-surprising-benefits-of-a-mid-career-break`
[5] `http://www.bbc.com/earth/story/20160823-the-real-origin-of-north-americas-wolves`
[6] `http://www.bbc.com/sport/cricket/37365637`
[7] `http://www.bbc.com/travel/story/20160912-the-treacherous-road-with-318-turns`

6.1.3 Questionnaires

Participants had to fill out four types of questionnaires. A pre-study questionnaire in the beginning, then for each customisation one questionnaire about the article and one about the customisation and at the end a post-study questionnaire. The reason why we also asked questions about the articles was on the one hand to encourage a more detailed viewing of the customisations and on the other hand to see if there are differences between the customisations in terms of conveying information.

All the questionnaires can be found in Appendix A.

Each questionnaire has a ”User ID” entry at the beginning to be able combine the results per participant at the end.

Although there were quite a lot of questions to answer for each participant, the cognitive load was lowered by the fact that the questions about the customisations were always the same, so participants got used to answering them. Also the fact that participants switched between reading on the chained displays, which also included walking around in the room, and sitting in front of the computer answering the questions prevented them from getting tired of the many questions. Some participants suggested that fewer questions would have been nice, but nobody mentioned it to be a problem in terms of concentration.

Participants were free to provide oral feedback whenever they liked. This was written down by us and later added to the appropriate feedback section.

Pre-Study

The pre-study questionnaire consisted of demographic questions and questions about the participants usage of public and semi-public displays.

Article

Each article had three individual questions. Two about the content of the text and one about the images to ensure people look at both, the text and the images.

Customisation

The customisation questions consisted of three parts.

First participants were asked on how many displays they concentrated their attention with this customisation as we wanted to see if this changes per customisation.

Second we asked them to fill out the System Usability Scale[6] (SUS) questionnaire. This is a well established questionnaire to analyse a system’s usability. We wanted to see if the customisations score differently with respect to usability. This questionnaire also allows the comparison of quite different systems in terms of usability, which benefits future research comparing their results to our setting. The SUS questions were used as they are without any changes.

As a third part, we asked participants to fill out an User Engagement Scale[21] (UES) questionnaire. This questionnaire explores the six dimensions of user engagement:

Focused Attention: This dimension describes how focused the user is on the task and if he blocks out things around him. Does the system attain the user's full attention?

Felt Involvement: High user engagement is also correlated with how involved the user feels with the system. Is it fun to use the system?

Novelty: Novelty describes how curious the user gets when using the system. Is he willing to explore the system longer than necessary to complete his task?

Endurability: Would the user use the system again? This depends on if the user sees his experience with the system as a success.

Aesthetic Appeal: Aesthetic Appeal describes the visual quality of the system. Do people like to look at it? Is it attractive?

Perceived Usability: This dimension includes the user's satisfaction with the system. Is it frustrating and demanding to use? It describes if the system is perceived as usable by the user.

We used this questionnaire to see how the customisations score with each UES dimension, as there might not be differences in user engagement in general but specific ones in certain dimensions. To keep the number of questions we ask on a reasonable level, but still cover all the dimensions we reduced the original 31 questions to 18 questions, with 2 to 4 questions per UES dimension. All the questions for the UES part were adapted to fit the terminology of viewing a chained display customisation.

Post-Study

After the participants had seen all the customisations, we asked them if they had seen them before this user study and if yes, how often. This was very important to verify how current chained display configurations are used and if our results are in sync with this. This is, if customisations which score high in usability and user engagement are also deployed more often than the other customisations.

Furthermore, we wanted the participants to rank the customisations for public and semi-public settings separately. They should also state what they liked most about the customisations (one specific customisation or a feature of some) and what they liked least. This would enable us to conduct further research and maybe refine or improve individual customisations or the customisation models.

6.2 Results

The study was carried out during 2 weeks and each participant took between 25 and 45 minutes to complete the study. Chocolate was provided as a thank-you gift.

Most of the analysis was done using *IBM SPSS Statistics Version 23*[8]. Data preprocessing

[8] https://en.wikipedia.org/wiki/SPSS

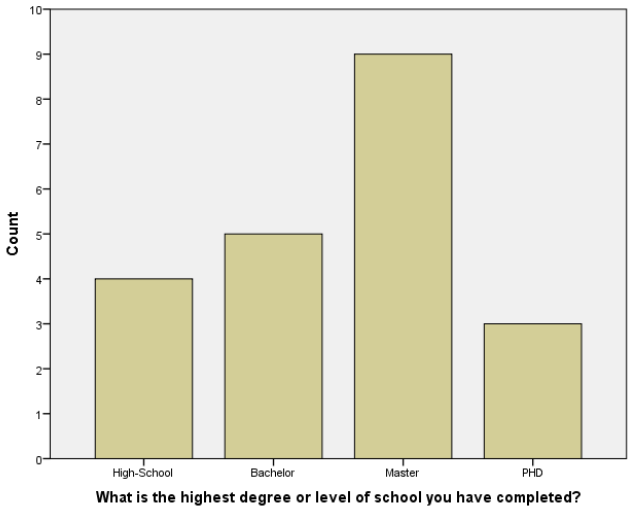


Figure 6.3: Educational degree of participants

and organizing was done with *Excel 2013*[9] and *Python 3.5.2*[10].

6.2.1 Participants

For the study we recruited 21 participants from our lab and our friends. Most of them were invited to the study by e-mail. We had 8 female and 13 male participants. Age ranged from 23 years to 55 years with a mean of 29.1 years. The 55 years old participant (User ID 19) was an extreme outlier with respect to the age. The distribution of their educational level can be seen in Figure 6.3.

11 of the participants worked or studied in the area of computer science. The rest was spread among different areas like economics, journalism, politics etc. 9 people had normal eyesight, 10 were wearing glasses or lenses and 2 had slightly bad eyesight, but nobody had problems reading the text on the displays.

6.2.2 Pre-Study

Figure 6.4 and Figure 6.5 show how often people look at public or semi-public displays. For public displays most people look at them on a daily or weekly basis. For semi-public displays the majority reported to look at them weekly or only once a month. All participants do look at public and semi-public displays, as no-one answered that he has never looked at such displays.

When asked on how many public or semi-public displays they look on a daily basis, most participants (13) responded with '1-5'. Nobody responded with 'more than 20'.

Only 'few' or even 'none' of those displays were in a chained configuration. And only 4

[9] `https://products.office.com/de-ch/excel`
[10] `https://www.python.org/`

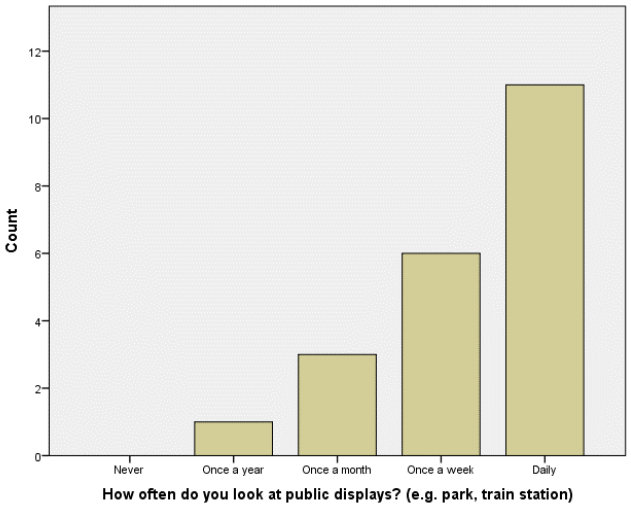


Figure 6.4: How often do people look at public displays?

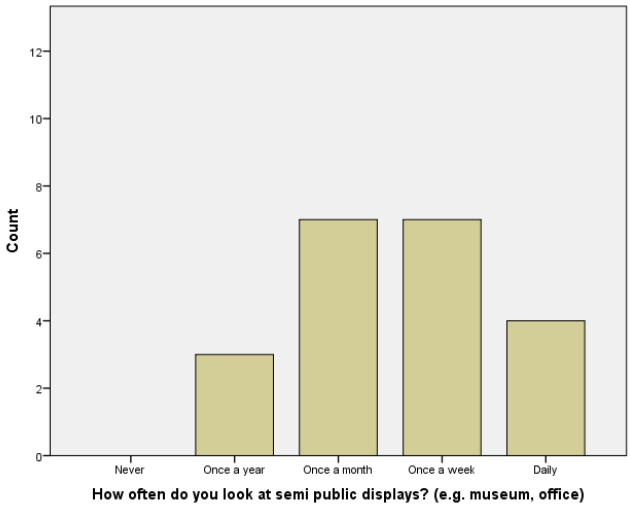


Figure 6.5: How often do people look at semi-public displays?

participants answered that 'some' or 'most' of the displays they see are in a chained configuration.

6.2.3 Article Questions

Quantitative Data

We analysed if the customisation had an effect on the amount of correctly answered article questions, but we found no statistical differences between the customisations. As this data was not normally distributed, we used a Friedman Test for the analysis.

Only very few people (0-2 per customisation) did know the content of the article before.

Qualitative Data

Participants reported their preferences for certain articles in the oral and written feedback. Most people liked the animal articles and disliked the cricket article. However, as the study was about customisations and not about the specific content, people provided more comments with regard to customisations.

6.2.4 Customisation Questions

General

Figure 6.6 shows on how many displays participants reported to have concentrated their attention, separated per customisations.

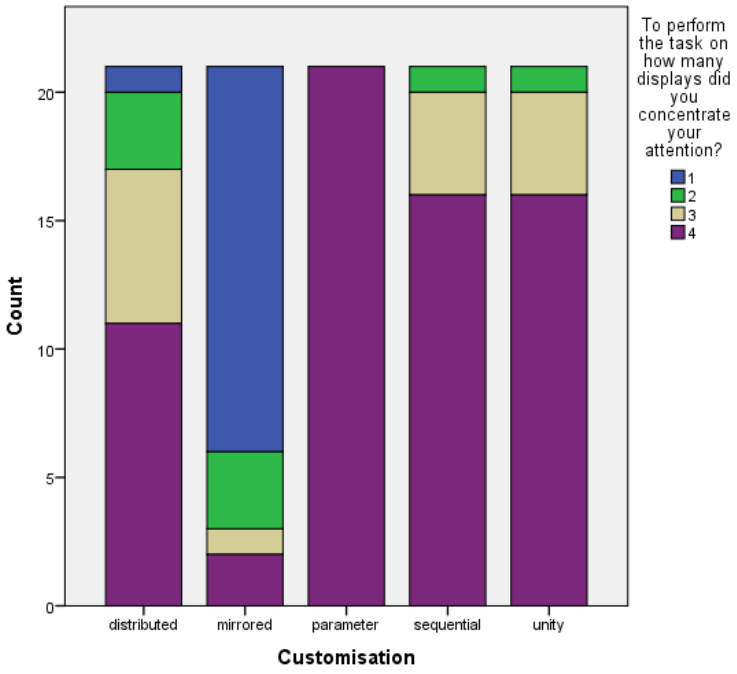


Figure 6.6: Chart: On how many displays did you concentrate your attention?

System Usability Scale

To calculate a single SUS Score from the answers to the ten questions of the System Usability Scale, we used the method described in the original paper [6].

We used an one-way ANOVA with repeated measures to determine statistical differences between the SUS Score means per customisation. To be able to use the one-way ANOVA with repeated measures, the data needs to be approximately normally distributed and the sphericity assumption should not be violated. These two assumptions were confirmed with a Shapiro-Wilk Test of Normality and Mauchly's Test of Sphericity.

Source		Type III Sum of Squares	df	Mean Square	F	Sig.	Partial Eta Squared
Customisation	Sphericity Assumed	13318.810	4	3329.702	21.624	.000	.520
	Greenhouse-Geisser	13318.810	3.431	3882.356	21.624	.000	.520
	Huynh-Feldt	13318.810	4.000	3329.702	21.624	.000	.520
	Lower-bound	13318.810	1.000	13318.810	21.624	.000	.520
Error (Customisation)	Sphericity Assumed	12318.690	80	153.984			
	Greenhouse-Geisser	12318.690	68.612	179.541			
	Huynh-Feldt	12318.690	80.000	153.984			
	Lower-bound	12318.690	20.000	615.935			

Table 6.1: Tests of Within-Subjects Effects for SUS mean scores per customisation

(I) Customisation	(J) Customisation	Mean Difference (I-J)	Std. Error	Sig.[b]	95% Confidence Interval for Difference[b]	
					Lower Bound	Upper Bound
Distributed	Mirrored	2.143	4.436	1.000	-11.847	16.132
	Parameter	-12.738*	3.937	.041	-25.154	-.322
	Sequential	-9.286	3.502	.153	-20.330	1.758
	Unity	19.524*	4.238	.002	6.158	32.889
Mirrored	Distributed	-2.143	4.436	1.000	-16.132	11.847
	Parameter	-14.881*	3.898	.011	-27.172	-2.590
	Sequential	-11.429*	3.285	.024	-21.787	-1.070
	Unity	17.381*	4.477	.009	3.263	31.499
Parameter	Distributed	12.738*	3.937	.041	.322	25.154
	Mirrored	14.881*	3.898	.011	2.590	27.172
	Sequential	3.452	2.479	1.000	-4.365	11.269
	Unity	32.262*	4.071	.000	19.424	45.100
Sequential	Distributed	9.286	3.502	.153	-1.758	20.330
	Mirrored	11.429*	3.285	.024	1.070	21.787
	Parameter	-3.452	2.479	1.000	-11.269	4.365
	Unity	28.810*	3.534	.000	17.666	39.953
Unity	Distributed	-19.524*	4.238	.002	-32.889	-6.158
	Mirrored	-17.381*	4.477	.009	-31.499	-3.263
	Parameter	-32.262*	4.071	.000	-45.100	-19.424
	Sequential	-28.810*	3.534	.000	-39.953	-17.666

Based on estimated marginal means

*. The mean difference is significant at the .05 level.

b. Adjustment for multiple comparisons: Bonferroni.

Table 6.2: Pairwise Comparison of the mean scores for SUS per customisation

The Tests of Within-Subjects Effects (Table 6.1) show that there are significant differences between the mean scores of the SUS per customisation ($F(4, 80) = 21.624, p < 0.001$).

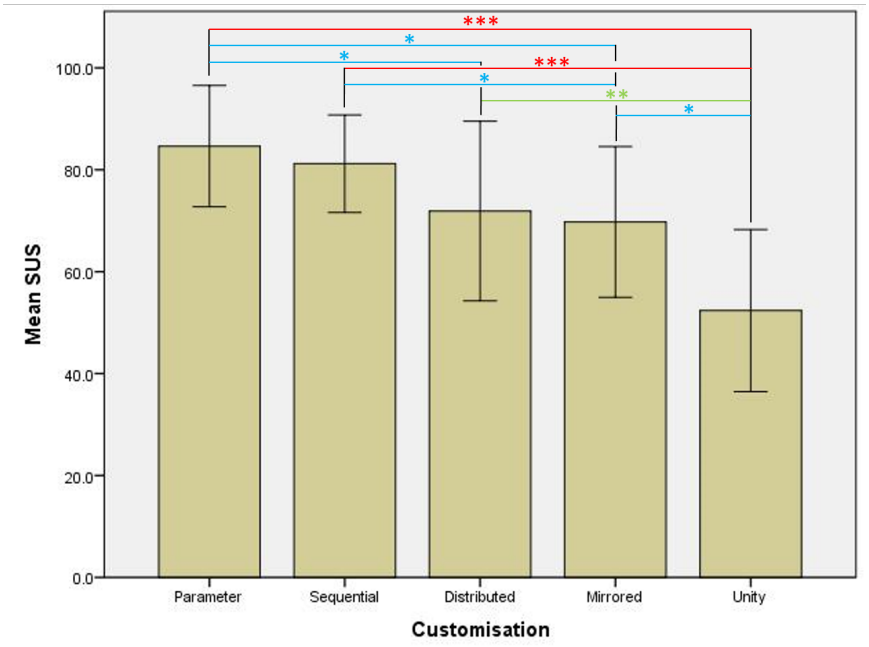
We therefore did a post-hoc analysis to compare the individual means. The results can be seen in Table 6.2. The yellow coloured lines are the ones which contain a significant result at the 0.05 level with a Bonferroni correction.

The descriptive statistics of the SUS Scores per customisation are in Table 6.3 and a corresponding bar chart in Figure 6.7. This bar chart also depicts at which p-values the differences between customisations are significant.

The participant with User ID 19 was a mild outlier in case of the SUS Score for the sequential customisation (SUS Score: 55.0). Since it was only a mild outlier and this participant was not an outlier when looking at the other customisations, we did not remove him from our analysis. The other SUS Scores per customisation did not have any outliers.

User Engagement Scale

To be able to analyse the Likert scale answers of the UES questions we defined a mapping, which can be seen in Table 6.4.



Differences significant at * $p < 0.05$, ** $p < 0.005$, *** $p < 0.001$

Figure 6.7: SUS Score Means per customisation bar chart

Customisation	Mean	Minimum	Maximum	Std. Deviation
Parameter	84.643	57.5	100.0	11.9186
Sequential	81.190	55.0	100.0	9.5727
Distributed	71.905	37.5	100.0	17.6229
Mirrored	69.762	37.5	92.5	14.7882
Unity	52.381	25.0	85.0	15.8996

Ordered by descending mean.

Table 6.3: Descriptive Statistics for the SUS means per customisation

Likert Scale Answer	Mapped Integer Value
Strongly Disagree	0
Disagree	1
Neutral	2
Agree	3
Strongly Agree	4

Table 6.4: Mapping of Likert Scale answers for UES questions to be able to do statistical tests.

Question	UES Dimension	Friedman Test	Customisation	Mean	SD	Median	Statistically significant Wilcoxon Signed Ranks Test[a]	
1. I lost myself in this viewing experience.	Focused Attention	$\chi^2(4) = 24.595$ $p < 0.001$	Sequential	2.14	0.91	2	Seq > Dis	< 0.01
			Parameter	2.10	1.00	2	Par > Dis	< 0.01
			Unity	1.81	1.12	1		
			Mirrored	1.22	1.11	1		
			Distributed	0.86	0.85	1		
2. I was so involved in my viewing task that I lost track of time.	Focused Attention	$\chi^2(4) = 8.561$ $p = 0.073$	Parameter	1.86	1.11	1	-	
			Sequential	1.62	1.02	1		
			Unity	1.57	1.21	1		
			Mirrored	1.19	0.81	1		
			Distributed	1.19	1.03	1		
3. I blocked out things around me when I was looking at the displays.	Focused Attention	$\chi^2(4) = 2.580$ $p = 0.630$	Sequential	2.48	0.87	3	-	
			Parameter	2.48	0.98	3		
			Distributed	2.48	1.08	3		
			Mirrored	2.14	1.06	3		
			Unity	2.10	0.89	2		
4. I was absorbed in my viewing task.	Focused Attention	$\chi^2(4) = 1.699$ $p = 0.791$	Sequential	2.57	0.81	3	-	
			Parameter	2.52	0.87	3		
			Distributed	2.33	1.02	3		
			Mirrored	2.19	1.08	3		
			Unity	2.19	1.08	3		
5. I felt involved in this viewing task.	Felt Involvement	$\chi^2(4) = 7.806$ $p = 0.99$	Parameter	2.76	0.89	3	-	
			Sequential	2.52	0.98	3		
			Distributed	2.24	0.83	2		
			Mirrored	2.14	0.91	2		
			Unity	2.10	1.04	2		
6. This viewing experience was fun.	Felt Involvement	$\chi^2(4) = 27.231$ $p < 0.001$	Parameter	2.86	0.91	3	Par > Uni	$= 0.01$
			Sequential	2.76	0.77	3	Seq > Uni	< 0.01
			Distributed	2.10	1.04	2		
			Mirrored	1.90	0.94	2		
			Unity	1.48	1.29	1		
7. I continued to look at the displays out of curiosity.	Novelty	$\chi^2(4) = 19.051$ $p = 0.001$	Parameter	2.62	0.97	3	Par > Mir	$= 0.03$
			Sequential	2.52	0.87	3	Seq > Mir	$= 0.01$
			Distributed	2.14	1.15	3		
			Unity	1.81	1.08	2		
			Mirrored	1.48	0.81	1		
8. The content of the displays incited my curiosity.	Novelty	$\chi^2(4) = 9.190$ $p = 0.057$	Parameter	2.95	0.74	3	-	
			Sequential	2.76	0.77	3		
			Unity	2.48	1.08	3		
			Distributed	2.33	1.28	3		
			Mirrored	2.19	1.17	2		
9. I felt interested in my viewing task.	Novelty	$\chi^2(4) = 7.217$ $p = 0.125$	Parameter	2.81	0.75	3	-	
			Sequential	2.76	0.70	3		
			Distributed	2.62	0.81	3		
			Unity	2.43	0.93	3		
			Mirrored	2.24	0.94	2		

a. The p-value from the Wilcoxon Signed Ranks Test was adjusted according to the Bonferroni correction: corrected p-value = normal p-value * 10 as we have 10 pairwise comparisons of customisations. The shown p-values are the corrected ones.

Table 6.5: UES Answers analysed per question (1-9)

Question	UES Dimension	Friedman Test	Customisation / Mean / SD / Median				Statistically significant Wilcoxon Signed Ranks Test[a]	
10. I consider my viewing experience a success.	Endurability	$\chi^2(4) = 18.458$ $p = 0.001$	Parameter	2.90	1.04	3	Par > Uni	$= 0.01$
			Sequential	2.86	0.91	3	Seq > Uni	< 0.01
			Distributed	2.43	0.98	2		
			Mirrored	2.24	1.26	2		
			Unity	2.00	0.95	2		
11. I would recommend looking at those displays to my friends and family.	Endurability	$\chi^2(4) = 28.603$ $p < 0.001$	Parameter	2.67	0.73	3	Par > Dis	$= 0.01$
			Sequential	2.62	0.97	3	Par > Mir	$= 0.03$
			Distributed	1.90	1.00	2	Par > Uni	< 0.01
			Mirrored	1.81	0.87	2	Seq > Mir	$= 0.03$
			Unity	1.43	1.08	1	Seq > Uni	< 0.01
12. This viewing configuration is attractive.	Aesthetic Appeal	$\chi^2(4) = 34.196$ $p < 0.001$	Parameter	2.95	0.81	3	Par > Mir	$= 0.02$
			Sequential	2.71	1.01	3	Par > Uni	< 0.01
			Distributed	2.05	1.07	2	Seq > Uni	< 0.01
			Mirrored	1.62	1.24	2		
			Unity	0.90	1.04	1		
13. I liked the graphics and images used on this customisation.	Aesthetic Appeal	$\chi^2(4) = 10.271$ $p = 0.036$	Sequential	3.00	0.71	3	Seq > Dis	$= 0.02$
			Parameter	2.81	0.68	3		
			Unity	2.62	0.92	3		
			Mirrored	2.52	0.87	3		
			Distributed	2.19	1.21	2		
14. The screen layout of this customisation was visually pleasing.	Aesthetic Appeal	$\chi^2(4) = 34.659$ $p < 0.001$	Parameter	2.71	0.78	3	Par > Uni	< 0.01
			Sequential	2.52	1.21	3	Seq > Uni	< 0.01
			Distributed	2.05	1.07	2	Dis > Uni	< 0.01
			Mirrored	1.81	1.03	2	Mir > Uni	$= 0.01$
			Unity	0.71	0.64	1		
15. I felt frustrated while looking at the displays.*	Perceived Usability	$\chi^2(4) = 34.821$ $p < 0.001$	Sequential	0.57	0.75	0	Seq > Mir	$= 0.02$
			Parameter	0.57	0.75	0	Seq > Uni	< 0.01
			Distributed	1.10	0.94	1	Par > Uni	< 0.01
			Mirrored	1.38	1.16	1	Dis > Uni	$= 0.03$
			Unity	2.29	1.19	3		
16. I found this customisation confusing to use.*	Perceived Usability	$\chi^2(4) = 34.006$ $p < 0.001$	Parameter	0.52	0.60	0	Par > Uni	< 0.01
			Sequential	0.62	0.81	0	Seq > Uni	< 0.01
			Mirrored	1.19	1.21	1	Mir > Uni	$= 0.02$
			Distributed	1.33	1.07	1	Dis > Uni	$= 0.03$
			Unity	2.52	1.08	3		
17. Looking at the displays was mentally taxing.*	Perceived Usability	$\chi^2(4) = 36.190$ $p < 0.001$	Parameter	0.86	0.79	1	Par > Uni	< 0.01
			Distributed	1.19	0.81	1	Dis > Uni	< 0.01
			Sequential	1.19	1.17	1	Seq > Uni	< 0.01
			Mirrored	1.24	1.04	1	Mir > Uni	< 0.01
			Unity	2.52	0.81	3		
18. This viewing experience was demanding.*	Perceived Usability	$\chi^2(4) = 32.570$ $p < 0.001$	Parameter	0.90	1.04	1	Par > Uni	< 0.01
			Sequential	1.05	1.02	1	Seq > Uni	< 0.01
			Distributed	1.10	0.94	1	Dis > Uni	< 0.01
			Mirrored	1.24	1.04	1	Mir > Uni	< 0.01
			Unity	2.62	0.67	3		

* indicates items that are reverse-coded

a. The p-value from the Wilcoxon Signed Ranks Test was adjusted according to the Bonferroni correction: corrected p-value = normal p-value * 10 as we have 10 pairwise comparisons of customisations. The shown p-values are the corrected ones.

Table 6.6: UES Answers analysed per question (10-18)

We analysed each UES question with a Friedman Test, as our data is ordinal, to determine if there are significant differences between the answers depending on the customisation. If the Friedman Test was significant, we continued the analysis for this question with a post-hoc Wilcoxon Signed Ranks Test. We chose this test, because our dependent variable (Likert scale answers) is ordinal and we do pairwise comparisons, so our independent variable (customisation) consists of two categorical. The p-values from the Wilcoxon Signed Ranks Test were adjusted to account for the testing of multiple hypothesis simultaneously (Bonferroni correction). The adjustment was to multiply the calculated p-values by 10 and only afterwards check if they are below the 0.05 threshold. We multiply by 10 as we compared 5 customisations with each other (4 + 3 + 2 + 1 = 10 comparisons). The results of this analysis can be seen in Table 6.5 (Questions 1 to 9) and Table 6.6 (Questions 10 to 18).

To see how the customisations behave with respect to the UES dimensions, we further analysed this data (Table 6.7). For each UES dimension we added all the answers per participant and customisation and did the same analysis as for the individual questions. By adding the answers they loose their meaning. This is, an 8.95 mean for Parameter with Focused Attention does not mean 'Agree' or something similar. However, we can use the relative results obtained when doing a Wilcoxon Signed Ranks Test to analyse the relative performance of the customisations depending on the UES dimension.

Qualitative Feedback

Participants reported (orally or in written form) that for the distributed customisation *"A visually more consistent representation (across screens) might be nice."*(P6). This would enable a more pleasant viewing experience and make the processing of the content easier. On the other hand participants liked the separation of the content per display as one user said: *"I liked most Parameter and Distributed, because a part of the topic is closed on one screen [so] I don't need to switch between screens."*(P18).

The mirrored customisation was reported as *"super boring"*(P14), because all displays showed the same content. Participants were also annoyed by the fact that they had to double check if the content is really the same to make sure they do not miss something. One person also mentioned that this customisation *"seems a bit as a waste of space"*(P9).

For the parameter customisation some mentioned that it is nice to have pictures and text on the same display (keeps motivation up to continue to the next display), while others found this fact mentally taxing and preferred a separation of pictures and text. It was also mentioned that it would be nice to have a *"visual hint, so it's clear that the four screens show different content from afar"*(P9).

For the sequential customisation the comments were similar to the ones for the parameter customisation. Some reported the separation of text and images as nice, while others preferred mixing those as *"reading text without pictures on large displays is demanding"*(P10). A clear indication for the direction of the text and that this text will continue on the next displays was also mentioned as a possible improvement as one person mentioned: *"I was first unsure if the text continues on the next screen and needed some time to realise [this]."*(P18).

As the unity customisation is the one which scored worst on the SUS Score, participants proposed the most improvements for this customisation. The most prominent one was to

UES Dimension	Friedman Test	Customisation	Mean	SD	Median	Statistically significant Wilcoxon Signed Ranks Test[a]	
Focused Attention	$\chi^2(4) = 9.679$ $p = 0.046$	Parameter	8.95	3.19	9	Par > Dis	$= 0.03$
		Sequential	8.81	2.91	10		
		Unity	7.67	3.29	8		
		Distributed	6.86	2.87	7		
		Mirrored	6.86	3.35	8		
Felt Involvement	$\chi^2(4) = 17.813$ $p = 0.001$	Parameter	5.62	1.60	6	Par > Uni	$= 0.01$
		Sequential	5.29	1.49	6	Seq > Uni	$= 0.01$
		Distributed	4.33	1.80	4		
		Mirrored	4.05	1.66	4		
		Unity	3.57	2.09	3		
Novelty	$\chi^2(4) = 20.202$ $p < 0.001$	Parameter	8.38	1.88	9	Seq > Mir	$= 0.03$
		Sequential	8.05	1.94	9	Par > Uni	$= 0.01$
		Distributed	7.10	2.90	9		
		Unity	6.71	2.53	7		
		Mirrored	5.90	2.34	6		
Endurability	$\chi^2(4) = 25.483$ $p < 0.001$	Parameter	5.57	1.54	6	Par > Uni	< 0.01
		Sequential	5.48	1.72	6	Seq > Uni	< 0.01
		Distributed	4.33	1.88	4		
		Mirrored	4.05	1.77	4		
		Unity	3.43	1.72	3		
Aesthetic Appeal	$\chi^2(4) = 29.707$ $p < 0.001$	Parameter	8.48	1.89	9	Par > Uni	< 0.01
		Sequential	8.24	2.51	9	Seq > Uni	< 0.01
		Distributed	6.29	3.00	7		
		Mirrored	5.95	2.78	6		
		Unity	4.14	1.95	4		
Perceived Usability*	$\chi^2(4) = 41.931$ $p < 0.001$	Parameter	2.86	2.31	3	Par > Uni	< 0.01
		Sequential	3.43	3.09	3	Seq > Uni	< 0.01
		Distributed	4.71	2.72	4	Dis > Uni	< 0.01
		Mirrored	5.05	3.75	5	Mir > Uni	< 0.01
		Unity	9.95	2.62	11		

* indicates items that are reverse-coded

a. The p-value from the Wilcoxon Signed Ranks Test was adjusted according to the Bonferroni correction: corrected p-value = normal p-value * 10 as we have 10 pairwise comparisons of customisations. The shown p-values are the corrected ones.

Table 6.7: UES Answers analysed per question (10-18)

not make characters cross the border between displays. *"I think if the breaks were in the middle of the sentences [instead of the characters] then everything was more pleasant and easier to read/follow."*(P14) Participants also said that a picture over the border might not be as mentally taxing as text over the border of multiple displays. Smaller seams would of course also help to solve this problem. *"It would probably have worked better if there was no broad margin between the displays"*(P6). On the positive side participants mentioned that the connection between the displays which is created by the unity customisation *"increased my motivation and attention to follow the information and move to different displays to find the rest of the information."*(P14)

6.2.5 Post-Study

Figure 6.8 shows a summary graph for the questions 'How often did you encounter the distributed/mirrored/parameter/sequential/unity customisation before?'. The corresponding statistics are shown in Table 6.8.

For ranking data of customisations for the public and semi-public setting, we created stacked bar charts to visualize the results (Figure 6.9 and Figure 6.10). The statistics are shown in Table 6.9 for the public setting and in Table 6.10 for the semi-public setting.

When asked which part of the chained display customisations they liked most, participants could not agree on a single answer. 6 people answered 'Sequential' which was the most common answer. Others mentioned 'Parameter' (4 people) or 'Distributed' (3 people). Some also mentioned that they liked it when the customisation does make use of the chained configuration by distributing the content among the different displays.

'Which part of the chained display customisations did you like least?' was answered with 'Unity' most often (7 people). Some specified that reading text over the border of two neighbouring displays was what they liked least. 3 people also mentioned that if the text over the border was removed the unity customisation may be much more pleasant. The mirrored customisation was mentioned 4 times to be least pleasing. Participants also mentioned the occasionally inconsistent design between the displays as annoying or the fact that not all customisations allow to grasp the content in one look.

General Comments

Participants said that it was sometimes hard for them to decide on the best customisation as this depends on the content on the displays as well as the aim, reason and location of the displays. While Mirrored ensures that everybody can read the content, the distributed customisation allows for a more diverse group of viewers. It was also mentioned that over time changing content would be nice.

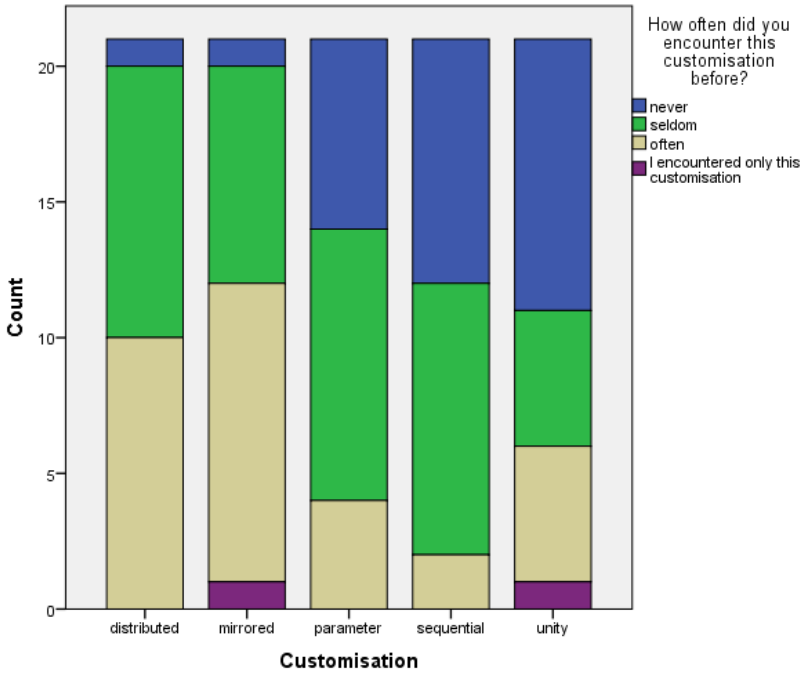


Figure 6.8: Chart: How often did participants encounter the different customisations before the study?

Customisation	Mean	Median	Mode
Distributed	1.43	1.00	1[a]
Mirrored	1.57	2.00	2
Parameter	.86	1.00	1
Sequential	.67	1.00	1
Unity	.86	1.00	0

a. Distributed had equally many votes for 1 and 2.

0 = never, 1 = seldom, 2 = often, 3 = I encountered only this customisation

Table 6.8: Statistics: How often did participants encounter the different customisations before the study?

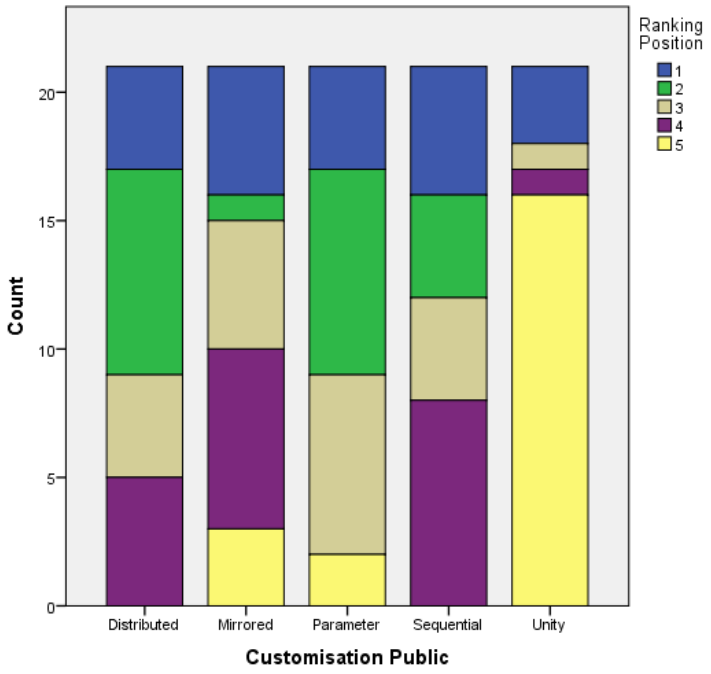


Figure 6.9: Chart: Ranking of customisations for a public setting

Customisation	Mean	Median	Mode
Distributed	2.48	2.00	2
Mirrored	3.10	3.00	4
Parameter	2.43	2.00	2
Sequential	2.71	3.00	4
Unity	4.29	5.00	5

Table 6.9: Statistics: Ranking of customisations for a public setting

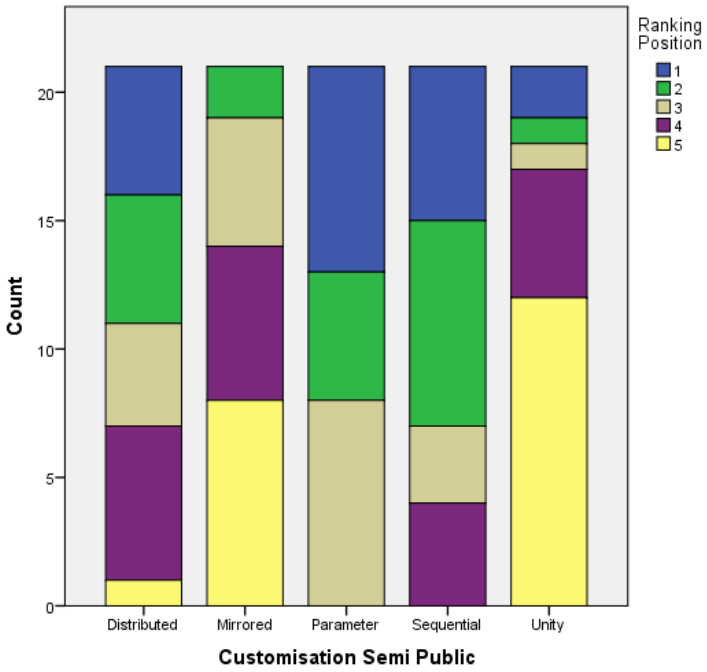


Figure 6.10: Chart: Ranking of customisations for a semi-public setting

Customisation	Mean	Median	Mode
Distributed	2.67	3.00	4
Mirrored	3.95	4.00	5
Parameter	2.00	2.00	1[a]
Sequential	2.24	2.00	2
Unity	4.14	5.00	5

Parameter was equally often placed on first and on third position

Table 6.10: Statistics: Ranking of customisations for a semi-public setting

6.3 Discussion

6.3.1 Participants

As we had a wide range of participants with regard to age, educational degree and occupation area, we believe that our results are representative and can be used by practitioners to improve the quality of their public display setting and by researcher to base their work on our findings.

6.3.2 Pre-Study

The pre-study questions confirmed that there are many public and semi-public displays in use today and people look at them on a regular basis. Public displays are more common than semi-public displays and are typically encountered on a daily basis. Many people encounter semi-public displays on a weekly basis. This underlines the relevancy of this work as public displays are clearly present in people's everyday lives. Although there are many public displays, only few of them are in a chained configuration, which supports the novelty of our research.

6.3.3 Article Questions

Since the customisation did not influence the amount of correctly answered article questions, we can conclude that the customisation does not influence the ability of viewers to grab the content. The customisation only has an effect on the engagement of the viewers and not the function of the displays to convey information.

The content (in our case: the article) has of course an effect on the viewers engagement as people prefer things they are more interested in, but since we randomised the combination of article and customisation for each participant, we eliminated this bias.

6.3.4 Customisation Questions

General

Figure 6.6 shows that when using the mirrored customisation viewers only concentrate their attention on one of the displays. For the distributed customisation it depends on which content the viewers like on how many displays they concentrate their attention. Only half of the participants concentrated their attention on all four displays with the distributed customisation. The parameter customisation is the only one where all participants concentrated their attention on all four displays. This is therefore the most absorbing customisation. Sequential and Unity also get the viewers to focus their attention on nearly all of the displays. Parameter, Sequential and Unity therefore make the viewers engage with the whole chained displays configuration, whereas Distributed and Mirrored promote a separation of the viewers among the displays.

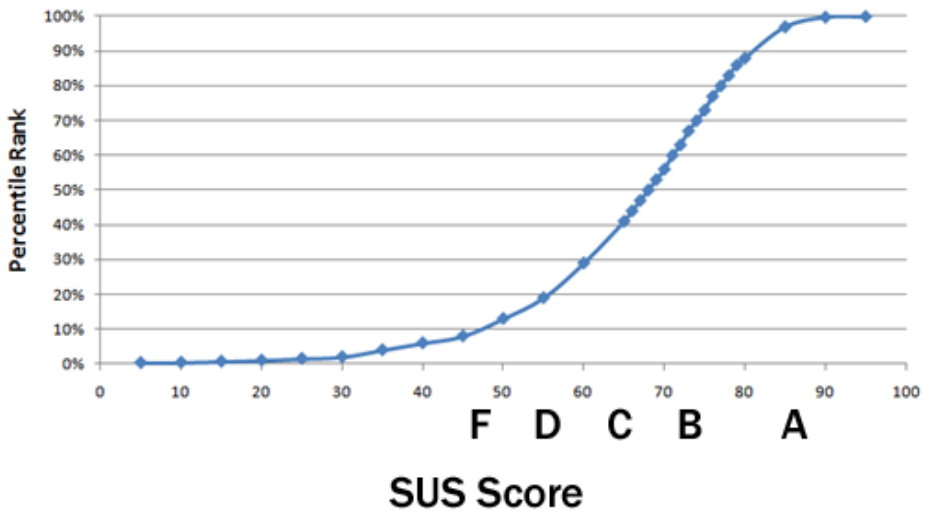


Figure 6.11: Graph to convert a SUS Score to a meaningful mark.

System Usability Scale

To interpret the SUS Scores we can use the method from Jeff Sauro [26]. Figure 6.11 shows his graph to convert SUS Scores into comparable marks.

Parameter would therefore be marked with an A, as well as Sequential. They both score excellent and are also not statistically different from each other. Distributed and Mirrored both score between B- and C+, which is an above average SUS Score. They are both statistically worse than Parameter. Mirrored also scores significantly worse than Sequential. Unity scores significantly worse than all the other customisations and is marked with a D-.

This ranking of the customisations by mean SUS Score reflects the overall result of this study. Parameter and Sequential are highly liked by the participants for their good use of the available display space and clear separation of the content for each display. The today's standard Distributed and Mirrored are generally good choices, but in most situations Parameter and Sequential would be a better fit. Unity is in general disliked by the viewers, but has some valid use cases.

User Engagement

When looking at the individual question tables (Table 6.5 and Table 6.6) we see that for each UES dimension, there exists at least one question for which the different customisations score significantly different. This supports the assumption that the customisations are different when looking at user engagement. Although Parameter and Sequential always are first or second when ordering by mean score and Distributed, Mirrored and Unity score third to fifth, the order is not the same across questions and dimensions. Each customisation does have varying strength and weaknesses when looking at the different UES dimensions.

Table 6.7 shows the analysis of the answers grouped by UES dimension. Again for each of the dimensions we do have significant differences between the customisations.

We will now discuss the results per UES dimension:

Focused Attention: The parameter customisation scores best and is significantly better than the distributed customisation. The others score similar, but it is important to mention that the unity customisation is third when ordering by mean score, which is its best result within the UES dimension analysis. So although it is disliked by the majority of people, it does focus the attention of the viewers. The results from the Focused Attention dimension are in line with what people answered when asked on how many displays they concentrated their attention (Figure 6.6). Parameter, Sequential and Unity spread the content over all the displays and therefore absorb the viewers' attention, while Distributed does a content switch per display which disturbs the attention of the viewer and allows for other things to steal his attention. With the Mirrored customisation people do not focus on more than one display and even get demotivated to focus on other displays when they see that they just show the same content.

Felt Involvement: Parameter and Sequential both score significantly better than Unity. Although Unity focuses the viewers' attention, they do not feel much involved. For Unity, in the individual questions, participants answered either with Disagree or with Neutral when asked about felt involvement. The reason for this is probably the mentally taxing and boring reading over the displays borders which opens up their mind for distractions. With the parameter and the sequential customisation people get drawn into the viewing experience. With the common topic for the displays in those cases, viewers can focus their mind on this topic and enjoy the viewing experience more intensively without any challenging reading tasks. For the distributed and mirrored customisation, people answered Neutral on average for the Felt Involvement questions as they might have problems being driven into the viewing experience when each display shows completely different or exactly the same content.

Novelty: Novelty describes the viewers curiosity evoked by the system. This is, of course, really low for the mirrored customisation. Viewers may be curious about the first display they see, but loose interest when they realise that the others show the same content. Unity was expected to score really well on novelty as few people have encountered it before. This was only partly confirmed. Most people answered Agree for the novelty questions, but Unity is still significantly worse than Parameter and does not have a significant higher score than any other customisation. The curiosity with the unity customisation was probably also dampened by the fact that people had to read text over the seams, which they reported as mentally taxing. Sequential and Parameter score very well on Novelty. Once more, the fact that with those customisations the displays share a common topic while presenting new or follow up content supports the viewers' curiosity.

Endurability: Participants would not recommend the unity customisation to their friends and family and are sure if they consider their viewing experience a success. As before, the reading over the seams is to blame, but it may also be difficult for them to reconstruct all the displays (as necessary for the unity customisation) in their head if they want to reflect on the viewing experience. This is much easier for the other customisations as they have a clear separation of the content per display. For Parameter and Sequential people answered that they would recommend them to their friends and family and also considered their viewing experience a success. Although the article questions showed that all customisation lead to a successful information gathering, it is important that the viewers also see their viewing experience as a success themselves, so they are more likely to return to those displays. Distributed and Mirrored were rated Neutral for Endurability, as their long-term benefit highly depends on

the setting and content shown.

Aesthetic Appeal: Parameter and Sequential are also with regard to Aesthetic Appeal the highest ranked customisations. The clear separation of content per display while still maintaining a visual link between them seems to also spike visual appeal. Distributed and Mirrored were ranked little less attractive than Parameter and Sequential, because the distributed customisation missed a visual common ground between the displays and the mirrored customisation had too much content packed on each display. The unity customisation and especially the text boxes over the large seams were not considered visually pleasing.

Perceived Usability: For this dimension the results are clearly distinct between the unity customisation and the other customisations. The Unity customisation was the only one which was reported to be frustrating to look at and to be mentally taxing. Participants said that this was because of the large seams between the displays. This made it challenging for them to read the text which crossed two displays. For all the other customisations participants negated the reverse coded questions of the Perceived Usability dimension, which means they score well on this dimension. The difference between Unity and all the other customisations is significant.

Qualitative Feedback

The qualitative feedback explains the results from the questionnaires quite well. The amount of criticism per customisation is in line with the SUS Score results (Table 6.3).

Participants mention that a visual commonality between the displays is important. This enables viewers to better understand what they are looking at, so they can faster and better decide on what they want to concentrate their attention on. Displays should also display clearly separated content parts, because people can then more easily focus on one display at a time and only need to move on to the next display when they have finished looking at the current one.

As Parameter and Sequential fulfil those two requirements, they score well throughout the questionnaires. People could not decide which of these two customisations do a better job at this as this also depends on the preferences of the individual viewers, but they agree that those requirements are important.

The mirrored customisation was mainly disliked, because people found it boring looking at the same content multiple times.

The distributed customisation lacked the visual commonality between the displays and was therefore also not so pleasant to use.

The unity customisation mainly suffered from the text over the display borders, but it was also mentioned that the unity customisation motivates to explore the other displays to get the whole picture.

6.3.5 Post-Study

Figure 6.8 shows how often participants encountered the different customisations before. And, as expected, the most prominent customisations are Mirrored and Distributed. Para-

meter, Sequential and Unity are also currently deployed in the wild, but much less frequently than Distributed and Mirrored. This backs up the importance of this work, as our results show that the currently most often deployed customisations (Distributed and Mirrored) are in general not optimal and Parameter and Sequential lead to much more user engagement and satisfaction and would therefore be a better choice. As one participant commented: *"I found myself liking the sequential [customisation] even though I have rarely seen a setup like it before."*(P12)

The ranking data for public settings (Figure 6.9) shows that there is no single best customisation. Although throughout the questionnaires Parameter and Sequential score best, people still ranked the other ones equally often first place. This is possibly because it is not quite clear what a public setting is. It can be a crowded train station in the morning or a nearly empty one at noon. But not only the place and number of people can vary for public settings, the shown content has also a large variability. They may show advertisement or public transport information or some news articles. All this diversity may lead to the fact that participants could not decide on a general ranking. However, they agreed more when we look at the second place. Distributed and Parameter score well, followed by Sequential and Mirrored. Unity scores worst. This shows that a clear separation of the content is important as well as the ability for the viewers to look at only one display and still get useful information while also being able to move to the other displays for new content.

When looking at the ranking for the semi-public case (Figure 6.10), we see that participants agreed much more on which customisation is a good fit and which one is not. This is explained by the fact that for a semi-public setting there are much fewer variations. Typically, there is not a large crowd in front of a semi-public display. What is more, the content is focused on other domains, such as time tables or news, in contrast to for example advertisement. The parameter customisation is ranked best, due to its prevalence in the lower ranks, and never having been ranked fourth or fifth. Parameter is followed by Sequential and Distributed. Those three can be considered an appropriate use of a chained display configuration in a semi-public space. The Mirrored and the Unity customisation are more often ranked on the last or second to last. This makes sense, as the mirrored customisation is not very useful in a semi-public space, except if it were really crowded. The Unity customisation again scores badly, because of the reading over seams and that people have to look at all the displays to understand the content.

The results from the questions 'Which part of the chained display customisation did you like most/least?' are similar to the ones from the semi-public ranking. Parameter, Sequential and Distributed were favoured by the participants, while Mirrored and Unity were in general disliked by the participants.

6.4 Summary

With this user study we have shown that the current approaches for customisations on public and semi-public displays are far from optimal. Participants confirmed that at the time of this writing the distributed and the mirrored customisation are most commonplace in reality, even though they generally do not provide a satisfactory solution. To use the distributed customisation makes sense if we have a diverse viewership, but otherwise some displays

waste space providing content most people are not interested in. The mirrored customisation is only useful if there are many simultaneous viewers. It ensures that everyone can read the content. In contrast, if we only have few viewers, they will feel annoyed and bored by the duplicated content on the other displays.

Although the best customisation depends on context, setting and content of the chained display configuration, participants agreed that the parameter and sequential customisation have good general properties. They have a visual commonality between the displays, they provide new content on each display and the content can be viewed on a per display basis, which makes it easier to grasp the content.

Those findings are backed by the fact that we had a diverse group of participants, and that we explored the participants engagement, satisfaction and critics with respect to the customisations with an in-depth analysis of the results.

7 Adapting to the viewers' walking direction

7.1 Motivation

The sequential customisation aligns the content to the order of the displays within the chained display configuration. The viewer, as mentioned before (Section 3.2.4), can therefore only enjoy the content when looking at the displays in the correct order. In our user study about user engagement with different customisations we displayed the sequential customisation from left to right, as this is the reading direction for the English language. In a real world scenario, however, the displays might be located such that some viewers encounter them from right to left, which prevents them from enjoying the content. A simple solution seems to be to adapt the order of the displays to the walking direction of the viewers. We tested how viewers would perceive such an adapting system with a small scale user study. In our user study we assume a single viewer for the displays and we adapt displays to his walking direction.

7.2 Setup

We used the three semi-public displays from our group to test how people perceive a chained display setup, which adapts to their walking direction. Figure 7.1 and Figure 7.2 show the setup. The social area and the printer room are three meters away from each other. The One News Template (Section 5.7) in the sequential customisation together with the The Guardian service (Section 5.5) were used for the study. Two Microsoft Kinect[1] sensors detected the walking direction of a single user in front of the displays. The system then changed the logical order of the displays depending on the walking direction of the viewer. Either the left

[1] `https://developer.microsoft.com/en-us/windows/kinect`, accessed January 16, 2017

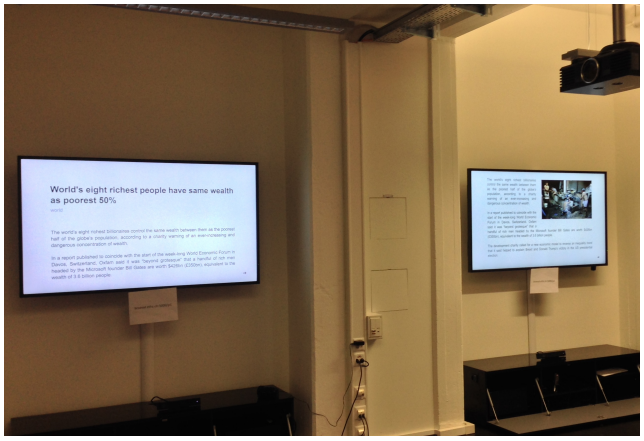


Figure 7.1: The two semi-public displays in the social area of our group.

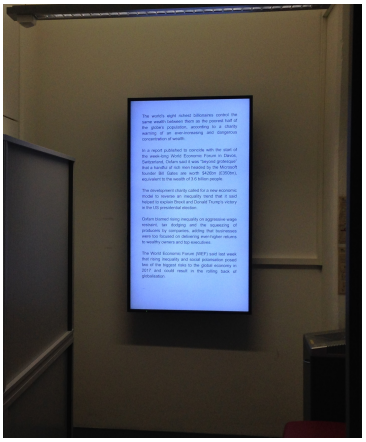


Figure 7.2: The semi-public display in the printer room of our group.

display in the social area was the first one (left to right) or the one in the printer room (right to left). Small arrows at the bottom of the screen of each display pointed into the walking direction of the user. This was added to give the user a hint in which direction the text will continue.

7.3 User Study

The user study took place on one day and we had six participants, which were either friends of us or people from the lab. Participants were asked first to walk in one direction through the corridor and then in the other direction, while looking at the displays. Afterwards they could further experiment with the system, before answering a questionnaire. The questions were asked by us and we also noted down the answers. The exact article on the displays did not matter for this user study. The displays showed most recent news and the participants could decide on their own how extensive they wanted to read the article.

7.3.1 Questions

The questionnaire for this user study consisted of the following six questions:

1. Which is your mother tongue's reading direction?
2. Do you like it, if the system adapts to your walking direction? Why?
3. Did you prefer one direction over the other? Which one and why?
4. Was the system easy to use?
5. Was the system easy to learn?

6. Do you consider the adaptive approach more efficient?

The reason to use the first question, was that we presumed that this might have an influence on the preferred direction of the viewer. The other questions targeted the performance and usefulness of an adaptive system over a static one.

7.3.2 Results

We were overwhelmed with positive feedback. Although one participant mentioned, that he does *" not see the use case for such a system."*(P6), all the others said they liked the system a lot (Question 2) and would be happy if more public displays adapted their logical order to the viewers' walking direction. The reason given, by the participants was that it is very important to see the title first instead of the end of the article. They also noted that they would not read the article at all if it had been displayed in the opposite way compared to their walking direction.

Two out of six participants grew up reading from right to left. The mother tongue's reading direction, however, had no influence on the preferred reading direction, but the reading direction of the shown text had. Three people stated that they preferred left to right, because this was also the reading direction of the text (in English) on the displays. One participant said, that the *"eyes are already moving from left to right. It is therefore also more pleasant to switch the displays from left to right."*(P1). Another participant, who does speak Farsi, mentioned: *"Right to left is probably more comfortable for languages, which one reads from right to left, like Farsi."*(P3). No one preferred viewing the displays from right to left. People either preferred left to right or said that both are equally good.

All participants said that the system was easy to use, as they simply need to walk by. *"The system seems invisible"*(P4), as there are no direct interactions needed for it to work.

The participants also agreed, that the system is easy to learn as it is *"immediately learned when encountering the second direction the first time."*(P4). One participant mentioned, however, that the system should better indicate, that it adapts itself to the viewer (P6).

The adaptive approach was considered more efficient than the static one by all participants, especially for semi-public settings where there are typically only a few simultaneous viewers (P2). It was mentioned by two participants, that this setup might not be more efficient for a multi user setting.

Participants also suggested the following improvements to the system:
- An indicator with the ID of the display the viewer is currently looking at (like: 2/3).
- A text like "Next Display" next to the arrow, to more clearly indicate that there is more to view.
- The One News Template changes the article from time to time. An indicator before the change would be nice, so viewers can estimate if it is worth to start reading (like: 20 seconds remaining).

7.3.3 Discussion

The feedback was really positive, and participants agreed that for a single viewer setting the adaptive approach is much easier to use and more efficient than the static approach. Which direction feels more natural does not depend on the viewers' mother tongue's reading direction, but on the reading direction of the displayed text. We can conclude, that it is important for chained displays in a sequential customisation to show the content in the order the viewers perceive it. By using methods to detect the walking direction of the viewers, this can for single viewer scenarios be sufficiently well done and leads to a high satisfaction of the viewers.

For multi viewer settings it still an open research question how a system should adapt its content to the viewers' walking direction or how else such a system should react to multiple viewers.

7.4 Summary

The user study showed, that adapting the content on a chained display configuration to the viewers' walking direction is very import to satisfy the viewers. Viewers do not pay attention to content which is displayed against their direction of movement. Although it is not clear how such a system should adapt itself in a multi viewer setting, it is for single viewers a necessary feature for a successful chained display deployment.

8

Design Guidelines for Chained Public and Semi-Public Displays

Based on the exploration of customisation models for chained public and semi-public displays (Chapter 3) and the results of our user studies (Chapter 6 and Chapter 7), we formulate the following design guidelines. Those design guidelines address practitioners and researchers who use and explore public or semi-public displays in a chained configuration. These guidelines describe what factors influence the performance of a customisation and which considerations should be taken when deploying a chained display configuration.

8.1 General

The content on the different displays should have a visual similarity. This is very important for the viewer of the displays. It eases his understanding of what he is looking at and perceive the chained displays as a whole.

The displays should clearly indicate what they display and what their relation to the other displays is. This helps the viewer to decide which display(s) he wants to look at in more detail and in which order.

Which customisation is best depends on:

8.2 The displays' stakeholders' intention

8.3 The distance between the displays

8.4 The order in which viewers encounter the displays

8.5 How much time the viewers are willing to spend in front of each display

8.6 The content

8.7 The viewers' expectations

8.8 The amount of viewers

8.9 The viewers' preferences

We will now explain those factors in more detail.

8.2 The Displays' Stakeholders' Intention

Each customisation has different effects on the user engagement. Depending on which dimension of the user engagement is important for the stakeholders, another customisation might be appropriate.

Focused Attention: Parameter focuses the viewers' attention more than Distributed. With Parameter, Sequential and Unity viewers will consider all displays. Using Distributed, this is unlikely as the viewer needs to be interested in all the content. With Mirrored the viewers focus their attention on only one of the displays.

Felt Involvement: Parameter and Sequential score better for Felt Involvement than Unity. Users agreed that the viewing experience was especially fun when exposed to those two.

Novelty: Parameter and Sequential feel most novel. Mirrored and Unity do not support the viewers' curiosity. Novelty is important to raise the viewers' interest beyond a quick glance.

Endurability: With Parameter and Sequential viewers are more likely to return to the displays. Display configurations with those customisations are also more likely to be recommended to friends and family by the viewers. With Unity only few see their viewing experience as a success, so most people will not return to the displays.

Aesthetic Appeal: The screen layout is most pleasing to viewers in the sequential and parameter customisations. Mirrored and Distributed score neutral on Aesthetic Appeal. Nice graphics can improve Unity's Aesthetic Appeal, but it is generally not considered attractive.

Perceived Usability: All customisations except Unity are perceived as highly usable. With Unity this depends on how the displayed application makes use of the unity customisation.

8.3 The Distance Between the Displays

Which customisation is preferable depends on the distance between the displays relative to the viewer's position. If the viewer easily can see and read the content of two displays at the same time from his position, the mirrored customisation does not make sense. It does, however, make sense to use the mirrored customisation if the displays are far away from each other and the viewer should be reminded of the content.

For the distributed and the parameter customisation, the distance between the displays is not relevant.

To use the sequential or the unity customisation in a meaningful way, the displays should not be too far away from each other, so viewers can still remember or at least partly see the content of the previous display.

8.4 The Order in Which Viewers Encounter the Displays

If the sequence in which the viewers will encounter the displays is not known, Parameter, Mirrored or Distributed are better suited than Sequential and Unity, as they do not impose any order on the content.

If the order is not random but either left to right or right to left, Unity is a good choice. For Sequential one needs to know the order of the displays as encountered by the viewers beforehand, to properly display the content. If, for a given point in time, all viewers see the displays in the same order, one can use detection methods (such as Microsoft Kinect[1]) to adjust the order of the sequential customisations to the viewers.

8.5 How Much Time the Viewers Are Willing to Spend in Front of Each Display

If the viewers will spend only a short amount of time per display, each display should of course only contain a small amount of information. If the amount of information per display needs to be large, then in this particular situation the mirrored customisation makes the most sense as the viewer can iteratively learn the information by looking at the same information multiple times.

If the viewers will spend a long time per display, the unity customisation is not well-suited. In this situation each display should be a closed entity on its own which the viewer can enjoy. This holds true for all customisations except Unity.

8.6 The Content on the Displays

The mirrored customisation is not suitable for a high amount of content, as it leads to small font and image sizes.

The distributed customisation needs the content to be unrelated and each piece of content should not be large either as it has to fit on one display.

With the sequential and the parameter customisation the content needs to be separable. If this is not the case, Unity should be used.

[1] `https://developer.microsoft.com/en-us/windows/kinect`, accessed January 16, 2017

8.7 The Viewers' Expectations

Typically, it is hard to know beforehand what the viewers will expect to see on certain public displays. Nevertheless it is important to think about it as this can also influence viewers' engagement. Fulfilling negative expectations or not living up to positive ones leads to boredom on the viewers' side and will prevent them from returning to the displays at another time.

The most risky customisation in this case is Mirrored. If people expect the other displays to show different content, they will be disappointed once they have realised the other ones show just the same content. They even might be upset as they have just wasted time and energy for this.

This effect is comparable with the distributed customisation when viewers expect follow-up content instead of unrelated content. Although they will not feel as negative as with the mirrored customisation, because they still get to see new content.

For the sequential and parameter customisation it is important to provide hints to the viewer that there are other displays and that they show related content. Otherwise a viewer might stop looking at the displays without getting the information he was looking for although it would have been on neighbouring displays. The unity customisation provides such a hint by design.

8.8 The Amount of Viewers

If there are many viewers in front of the chained displays, not everyone will be able to see all the displays. If it is critical to get the content across to everyone, the mirrored customisation is most suited. Otherwise, Distributed and Parameter are a good choice. Both customisations allow people to choose the display in whose content they are most interested.

If there will be only a view people in front of the displays at the same time, all customisations except the mirrored one should be used.

8.9 The Viewers' Preferences

Which customisation maximizes the individual viewers' engagement also depends on their individual preferences. This simply means that they might prefer one customisation over the because of their personal favour. This is hard to account for in a public setting. But in a semi-public setting one might know enough about the viewers to incorporate their preferences.

9 Conclusion

In this thesis we explored the customisation of chained public and semi-public displays and provided guidelines for choosing an appropriate customisation for a given chained display configuration.

We looked into how content presentation can make use of a chained display configuration by examining currently used techniques and additionally by organizing a brainstorming session within our lab. Those results where then further analysed to create customisation models which represent key features of different customisation strategies of chained display configurations.

Those customisation models are:

Distributed: Each display shows a different application.

Mirrored: All displays show the same content.

Parameter: Each display shows the same kind of content, but different realisations depending on a parameter which changes from display to display.

Sequential: The displays have a strict order imposed by the shown content. This creates a flow and guides the viewer from display to display.

Unity: All displays form one large logical display, which ignores physical borders of the individual displays.

We further categorise them into two groups: Customisations showing unrelated content (Distributed) and customisations which display related content on each display (Mirrored, Parameter, Sequential, Unity). Unrelated customisations show content on each display which is not related to the content on the other displays. Related customisations show content which is related to each other on the displays.

Based on those customisation models, we extended the ScreenPress PDS to provide support for chained display configurations. We depict the necessary features to support chained display configurations as well as an example GUI in form of a Wordpress plug-in for ScreenPress.

Furthermore, we developed some sample applications. These applications demonstrate how one can build applications for chained public displays using our extension to the ScreenPress system. They also provide an example of content customisation on chained displays in different domains.

We then evaluated the customisation models by means of a user study. This study provided valuable insights into the effect of the different customisations on system usability and user engagement. The customisations differ in usability as well as with respect to different dimensions of user engagement. However, they all convey information equally well. The study also showed, that, the at this time most commonly deployed customisations, Distributed and Mirrored are generally not the most effective choice and that Parameter or Sequential provide a much better experience to the viewer. Viewers want a commonality between the displays of a chained display configuration with regard to both the visual presentation and the content. However, simply repeating the content was reported to be boring and slightly changing content leads to a higher engagement of the viewers.

Another user study showed, that adapting the logical ordering of the displays to the viewers' walking direction is important when using the sequential customisation.

A summary of the results of our studies can be found in the design guidelines we provide to practitioners and researchers who plan to deploy chained public or semi-public displays.

9.1 Limitations

Although the results from our first user study provide a reasonable picture of how customisations on chained public displays effect viewers, the study was limited to a lab environment. This provided us with detailed insights in how viewers interact with a chained display configuration. However, to truly validate the results, one needs to investigate the effects of the customisations outside of the lab, in a real life deployment. This was, however, out of the scope of this thesis, as in the wild studies are, especially in the field of public displays, highly complex. One would ideally investigate an in the wild deployment at multiple locations, which differ regarding the amount of people or the walking direction of the viewers, as those probably have a high influence on the performance of the customisations.

An in the wild study would naturally also explore the effect of multiple viewers simultaneously looking at the system, whereas we only investigated single viewer behaviour. Since this is the start of research on customisations of chained public displays, we focused on the most simple case which is single viewer in a lab environment. This provides the rough direction in which we encourage further research to go. In particular one such study should look at the effects the different customisations have on multiple simultaneously viewers.

9.2 Future Work

In this section we will highlight the most promising directions for future research. There are of course endless possibilities for further investigation of customisations of chained public displays, but we think that the following topics will certainly provide valuable insights.

Interactive displays are, as mentioned in the beginning of this thesis, a highly active field of research. Combining our customisation models with interactive displays raises a couple of new, exciting challenges, which could be topics of future research. It is not obvious how for instance a sequential customisation should behave if there are viewers interacting with the displays. In case of the mirrored customisation this is probably more straight forward. Working with interactive displays might even yield more customisation models not possible with broadcast displays.

Our study revealed that depending on the situation some customisations perform better than others in terms of user engagement. How do single large displays compare to this? Some research highlighted the benefits of small displays over large ones. In what cases a single large display is beneficial compared to a chained configuration with a proper customisation and vice versa remains to be seen. Chained displays may combine the best of small and large displays. Viewers can split among displays and enjoy the content in a more private fashion, while multiple displays still provide a lot of screen space to display content.

The number of displays in a chained display configuration is not fixed. We encountered configurations ranging from two to eight displays. It would be interesting to see how the number of displays in a chained public display setup influences user engagement. We presume this likely depends on the customisation. The effect of the mirrored customisation might not be affected by the number of displays, while Unity might suffer from too many displays. The sequential customisation probably benefits from more displays as the viewers might feel more involved into the viewing experience when looking at a longer chain of sequential displays.

A

User Study Questionnaires

Questions marked with an * were mandatory.

A.1 Pre-Study Questionnaire

Page 1: General Info

1. User ID*

Page 2: Demographics

2. What is your year of birth?*
3. What is your gender?* Female | Male
4. What is the highest degree or level of school you have completed?* School | High-School | Bachelor | Master | PHD | Other
5. What is your occupation area?*
6. Do you have normal eyesight?* Yes | No
7. Do you wear glasses or lenses?* Yes | No

Page 3: Public displays

1. How often do you look at public displays? (e.g. park, train station)* Never | Once a year | Once a month | Once a week | Daily
2. How often do you look at semi public displays? (e.g. museum, office)* Never | Once a year | Once a month | Once a week | Daily
3. How many public or semi public displays do you encounter per day?* 0 | 1-5 | 6-10 | 11-20 | more than 20

4. How many of those are chained display configurations?*
   none | few | some | most | all

A.2 Article Questionnaire

Page 1: General Info

1. User ID*
2. Customisation* Mirrored | Parameter | Sequential | Unity | Distributed

Page 2: Article Questions

3. Did you know the content of the article previous to this study?* Yes | No | Maybe
4. to 6. are article specific
7. Do you have any comments about this article?

Giraffe

4. How many giraffe species are there? (0 = I don't know)*
5. For roughly how long have these species been separated? (0 = I don't know)*
6. How many giraffes were in the image with the most giraffes? (0 = I don't know)*

Sabbatical

4. How long does Winston stay away from his job? (0 = I don't know)*
5. In how many pictures were people visible? (0 = I don't know)*
6. In which year was the book 'Reboot Your Life' published? (0 = I don't know)*

Wolves

4. About which wolf species does the article mainly talk? (0 = I don't know)*
5. According to the article what do wolves symbolize? (0 = I don't know)*
6. How many of the pictures contain snow? (0 = I don't know)*

Cricket

4. When could this new competition possibly start? (0 = I don't know)*
5. How many of the pictures highlight a single player? (0 = I don't know)*

6. How many teams will participate in this competition? (0 = I don't know)*

Motorcycle

4. In which country is the road? (0 = I don't know)*
5. What is the roads nickname? (0 = I don't know)*
6. On how many pictures is the road visible? (0 = I don't know)*

A.3 Customisation Questionnaire

Page 1: General Info

1. User ID*
2. Customisation* Mirrored | Parameter | Sequential | Unity | Distributed
3. Article* Giraffe | Sabbatical | Wolves | Cricket | Motorcycle

Page 2: About the task

4. To perform the task on how many displays did you concentrate your attention?* 0 | 1 | 2 | 3 | all

Page 3: Customisation - System Usability Scale
Strongly disagree | Disagree | Neutral | Agree | Strongly agree

5. I think I would like to use the system frequently*
6. I found the system unnecessarily complex*
7. I thought the system was easy to use*
8. I think that I would need a technical person to be able to use this system*
9. I found the various functions in this system were well integrated*
10. I thought there was too much inconsistency in this system*
11. I would image that most people would learn to use this system*
12. I found the system very cumbersome to use*
13. I felt very confident using the system*
14. I need to learn a lot of things before I could get going with this system*

Page 4: Customisation - User Engagement Scale
Strongly disagree | Disagree | Neutral | Agree | Strongly agree

15. I lost myself in this viewing experience.*
16. I was so involved in my viewing task that I lost track of time.*
17. I blocked out things around me when I was looking at the displays.*
18. I was absorbed in my viewing task.*

19. I felt involved in this viewing task.*
20. This viewing experience was fun.*
21. I continued to look at the displays out of curiosity.*
22. The content of the displays incited my curiosity.*
23. I felt interested in my viewing task.*
24. I consider my viewing experience a success.*
25. I would recommend looking at those displays to my friends and family.*
26. This customisation is attractive.*
27. I liked the graphics and images used on this customisation.*
28. The screen layout of this customisation was visually pleasing.*
29. I felt frustrated while looking at the displays.*
30. I found this customisation confusing to use.*
31. Looking at the displays was mentally taxing.*
32. This viewing experience was demanding.*

Page 5: Comments

33. Do you have suggestions or comments regarding this customisation?

A.4 Post-Study Questionnaire

Page 1: General Info

1. User ID*

Page 2: Debriefing

2. How often did you encounter the distributed customisation before?
   never | seldom | often | I encountered only this customisation
3. How often did you encounter the mirrored customisation before?
4. How often did you encounter the parameter customisation before?
5. How often did you encounter the sequential customisation before?
6. How often did you encounter the unity customisation before?
7. Please rank the customisations for a public setting (e.g. park, train station) (1 best - 5 worst)*
8. Please rank the customisations for a semi public setting (e.g. museum, office) (1 best - 5 worst)*
9. Which part of the chained display customisations did you like most?*
10. Which part of the chained display customisations did you like least?*
11. General comments or suggestions for improvement?

B

How to install ScreenPress and the Chained Display Extension

B.1 Installation of ScreenPress

- Install WTServer[1] (or a similar server stack like MAMP[2] or XAMPP[3])
- Download PHP 5.5.*x*.zip[4], where *x* depends on the newest available version
  - extract all
  - `cp php-5.5.x/bin/php-5.5.x /WTServer/bin/PHP/32bit-PHP-5.5.x`
- install the newest version of node.js[5].
- copy the `wordpress` folder to `c:/WTServer/WWW/`
- set the document root of your server (e.g. nginx) to the wordpress folder (e.g. `c:/WTServer/WWW/wordpress`)
  - if you are using nginx: enable wordpress URL rewriting by changing

```
location / {
    index            index.html index.php;
}
```

[1] https://wtserver.wtriple.com/
[2] https://www.mamp.info/
[3] https://www.apachefriends.org/
[4] https://sourceforge.net/projects/wtnmp/files/extras/
[5] https://nodejs.org/en/

to

```
location / {
    index            index.html index.php;
    try_files $uri $uri/ /index.php?q=$request_uri;
}
```

in `nginx.conf`

- adapt `DB_USER` and `DB_PASSWORD` in `wordpress/wp-config.php` to your mysql setting
- set `var serverURL` in `wordpress/xdserver/app.js` and in `wordpress/wp-content/themes/twentyfourteen/ambiJson.php` to `yourdomain.com`
  - optional: if you use other files than `ui.html` or `interactor.html` in `wordpress/xdserver/`, you need to make sure they contain `<!--PLACEHOLDER: server url-->`, so that the Front Server can insert its own URL at these places when serving those files.
- import the `wordpress.sql` database
  - If you use nginx: set `client_max_body_size` to `100M` in `nginx.conf` in the http > server configuration.
  - use phpMyAdmin to create new database called `wordpress`
  - select this database and import `wordpress.sql`
- set the wordpress url (siteurl and home) inside the wordpress database (table wp_options) to `yourdomain.com`
- if needed: adapt `wordpress.conf` and `xdserver.conf` in `/c/WTServer/conf/domains.d/`

B.2 Installing the Chained Display Extension

- If not yet installed: install the **ScreenPress Chained Display Extension** wordpress plugin by moving the `screenpress-chained-display-extension` folder into `wordpress/wp-content/plugins/` and activating it through the wordpress GUI.
- install services (those are wordpress plugins starting with **ScreenPress App - ...**) by moving them into `wordpress/wp-content/plugins/` and activating them through the wordpress GUI.
- install templates by moving their folder into `wordpress/xdserver/templates/`

B.3 Running the PDS

- start WTServer

```
cd /c/WTServer/
WTServer --phpCgiServers=25
```

- start node.js server

```
cd /c/WTServer/WWW/wordpress/xdserver
node app.js
```

B.4 Useful links

- visit
  `yourdomain.com`
  to get to the wordpress interface
- visit
  `yourdomain.com:5000/?display=E%20106%20TV`
  for the view on the display *"E 106 TV"*.
- visit
  `yourdomain.com:5000/interactor/?display=E%20106%20TV`
  to tell the display *"E 106 TV"* which service to show.
- visit
  `yourdomain.com/ambimajson/?start=yes&displayName=E%20106%20TV&ptype=services&context={}`
  to get a list of the activated services for a given context for the display *"E 106 TV"*.

List of Figures

List of Tables

List of Code Snippets

Acknowledgements

Thank you, Prof. Dr. Moira Norrie and Amir E. Sarabadani Tafreshi for making this thesis possible. This research project provided me with lots of interesting and new experiences.

Thank you, Amir E. Sarabadani Tafreshi for your guidance, input and feedback through out my Master Thesis.

Thank you, Dr. Amirehsan Sarabadani Tafreshi for helpful tips and comments on the statistical analysis of the user study results.

I would like to thank the participants of the brainstorming session for their great input.

I would also like to thank all the participants of the user study for their time and feedback.

Special thanks go to the other students in the GLOBIS group who also worked on public displays for lots of valuable discussions on the topic.

And last but not least I would like to thank the ones who proofread this thesis.

Bibliography

[1] Florian Alt. *A Design Space for Pervasive Advertising on Public Displays*. Institut fuer Visualisierung und Interaktive Systeme der Universitaet Stuttgart, 2012.

[2] Florian Alt, Nemanja Memarovic, Ivan Elhart, Dominik Bial, Albrecht Schmidt, Marc Langheinrich, Gunnar Harboe, Elaine Huang, and Marcello P Scipioni. Designing shared public display networks–implications from todays paper-based notice areas. In *International Conference on Pervasive Computing*, pages 258–275. Springer, 2011.

[3] Florian Alt, Stefan Schneegaß, Albrecht Schmidt, Jörg Müller, and Nemanja Memarovic. How to evaluate public displays. In *Proceedings of the 2012 International Symposium on Pervasive Displays*, page 17. ACM, 2012.

[4] Xiaojun Bi, Seok-Hyung Bae, and Ravin Balakrishnan. Effects of interior bezels of tiled-monitor large displays on visual search, tunnel steering, and target selection. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 65–74. ACM, 2010.

[5] John Brooke. Sus: a retrospective. *Journal of usability studies*, 8(2):29–40, 2013.

[6] John Brooke et al. Sus-a quick and dirty usability scale. *Usability evaluation in industry*, 189(194):4–7, 1996.

[7] Moira Burke, Anthony Hornof, Erik Nilsen, and Nicholas Gorman. High-cost banner blindness: Ads increase perceived workload, hinder visual search, and are forgotten. *ACM Transactions on Computer-Human Interaction (TOCHI)*, 12(4):423–445, 2005.

[8] Nigel Davies, Marc Langheinrich, Rui Jose, and Albrecht Schmidt. Open display networks: Towards a new communications medium for the 21st century. *IEEE Computer*, 45(5):58–64, 2012.

[9] Martin J Eppler and Jeanne Mengis. The concept of information overload: A review of literature from organization science, accounting, marketing, mis, and related disciplines. *The information society*, 20(5):325–344, 2004.

[10] Jonathan Grudin. Partitioning digital worlds: focal and peripheral awareness in multiple monitor use. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 458–465. ACM, 2001.

[11] Elaine M Huang, Anna Koster, and Jan Borchers. Overcoming assumptions and uncovering practices: When does the public really look at public displays? In *International Conference on Pervasive Computing*, pages 228–243. Springer, 2008.

[12] Elaine M Huang and Elizabeth D Mynatt. Semi-public displays for small, co-located groups. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 49–56. ACM, 2003.

[13] Marko Jurmu, Leena Ventä-Olkkonen, Arto Lanamäki, Hannu Kukka, Netta Iivari, and Kari Kuutti. Emergent practice as a methodological lens for public displays in-the-wild. In *Proceedings of the 5th ACM International Symposium on Pervasive Displays*, pages 124–131. ACM, 2016.

[14] Andol X Li, Xiaolong Lou, Preben Hansen, and Ren Peng. Improving the user engagement in large display using distance-driven adaptive interface. *Interacting with Computers*, page iwv021, 2015.

[15] Jock D Mackinlay and Jeffrey Heer. Wideband displays: mitigating multiple monitor seams. In *CHI'04 extended abstracts on Human factors in computing systems*, pages 1521–1524. ACM, 2004.

[16] Nemanja Memarovic, Ivan Elhart, and Elisa Rubegni. Developing a networked public display system. *IEEE Pervasive Computing*, 15(3):32–39, 2016.

[17] Jörg Müller, Robert Walter, Gilles Bailly, Michael Nischt, and Florian Alt. Looking glass: a field study on noticing interactivity of a shop window. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 297–306. ACM, 2012.

[18] Jörg Müller, Dennis Wilmsmann, Juliane Exeler, Markus Buzeck, Albrecht Schmidt, Tim Jay, and Antonio Krüger. Display blindness: The effect of expectations on attention towards digital signage. In *International Conference on Pervasive Computing*, pages 1–8. Springer, 2009.

[19] Heather L. O'Brien and Paul Cairns. An empirical evaluation of the user engagement scale (ues) in online news environments. *Information Processing & Management*, 51(4):413–427, 2015.

[20] Heather L. O'Brien, Luanne Freund, and Richard Kopak. Investigating the role of user engagement in digital reading environments. In *Proceedings of the 2016 ACM on Conference on Human Information Interaction and Retrieval*, pages 71–80. ACM, 2016.

[21] Heather L. O'Brien and Elaine G Toms. The development and evaluation of a survey to measure user engagement. *Journal of the American Society for Information Science and Technology*, 61(1):50–69, 2010.

[22] Heather L. O'Brien and Elaine G Toms. Examining the generalizability of the user engagement scale (ues) in exploratory search. *Information Processing & Management*, 49(5):1092–1107, 2013.

[23] Timo Ojala, Vassilis Kostakos, Hannu Kukka, Tommi Heikkinen, Tomas Linden, Marko Jurmu, Simo Hosio, Fabio Kruger, and Daniele Zanni. Multipurpose interactive public displays in the wild: Three years later. *Computer*, 45(5):42–49, 2012.

[24] Florian Ott, Andrea Nutsi, and Peter Lachenmaier. Information ergonomics guidelines for multi-user readability on semi-public large interactive screens. In *Proc. of the Workshop Information Ergonomics at iKnow Conference*, 2014.

[25] Peter Peltonen, Esko Kurvinen, Antti Salovaara, Giulio Jacucci, Tommi Ilmonen, John Evans, Antti Oulasvirta, and Petri Saarikko. 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*, pages 1285–1294. ACM, 2008.

[26] Jeff Sauro. Measuring usability with the system usability scale (sus), 2011. [Online; Stand 10. November 2016].

[27] Oliver Storz, Adrian Friday, Nigel Davies, Joe Finney, Corina Sas, and Jennifer Sheridan. Public ubiquitous computing systems: Lessons from the e-campus display deployments. *IEEE Pervasive Computing*, 5(3):40–47, 2006.

[28] Constantin Taivan, Rui José, and Bruno Silva. Web-based applications for open display networks: developers perspective. *International journal of computer systems science and engineering*, 30(1):21–30, 2015.

[29] Maurice Ten Koppel, Gilles Bailly, Jörg Müller, and Robert Walter. Chained displays: configurations of public displays can be used to influence actor-, audience-, and passer-by behavior. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 317–326. ACM, 2012.

[30] Mark Weiser and John Seely Brown. The coming age of calm technology. In *Beyond calculation*, pages 75–85. Springer, 1997.

[31] Eric N Wiebe, Allison Lamb, Megan Hardy, and David Sharek. Measuring engagement in video game-based environments: Investigation of the user engagement scale. *Computers in Human Behavior*, 32:123–132, 2014.