The Many Faces of Consistency in Cross-Platform Design: A Whitepaper

2 The Many Faces of Consistency in Cross-Platform Design: A Whitepaper

Jeffrey Nichols, Carnegie Mellon University, 5000 Forbes Avenue, Pittsburgh, PA 15213-3891, USA, jeffreyn@cs.cmu.edu

Kai Richter, Computer Graphics Center (ZGDV), Fraunhoferstr. 5, 64283 Darmstadt Germany, kai.richter@zgdv.de

Krzysztof Gajos, University of Washington, 370 Allen Center, Seattle, WA 98195 USA, kgajos@cs.washington.edu

This white paper summarizes the position papers and questionnaires filled out by the participants of "The Many Faces of Consistency in Cross-Platform Design" workshop, to be held at CHI'2006. The whitepaper will serve as additional background and hopefully lay the framework for our discussions in the workshop. The goal is to synthesize the opinions from the position papers and questionnaires to give us all a clear idea of where we are all in agreement, where we differ substantially, and to identify any areas of discussion in which participants have no interest.

INTRODUCTION

"The environment should be a coherent user experience." (Trapp and Schmettow 2006)

A goal of this workshop is to advance our understanding of how cross-platform interfaces, which hopefully will soon be a large part of our environments, can offer a coherent user experience. From the title of the workshop, it is clear that we believe some portion of this goal can be addressed by creating cross-platform interfaces that are "consistent." Part of this workshop will define what consistency is, and another part will address the complex factors and interdependencies that lead to consistency and also usability.

We decided to write this white paper in advance of the workshop to bring together the many different perspectives that we found in your position papers and to gain a better understanding of the base that our workshop will build upon. The white paper also gives us another opportunity to present results from the questionnaire that we distributed in advance of the workshop. Of particular interest from the questionnaire are the definitions of cross-platform consistency, which are discussed below.

To understand your opinions in the context of the workshop, we started with the initial four questions asked in our workshop proposal:

1. Dimensions: What are the different dimensions along which consistency can be ensured, how are they related, and what is their relative importance? Past research has identified a number of dimensions of consistent design on a single platform, however there is no integrating vision or framework available defining and putting into relation the specific dimensions of cross-device consistency. Interrelations and trade-offs have to be identified and discussed. One possible result of this workshop would be to formulate such an integrating framework.
2. Unique Problems: What are the unique problems for ensuring consistency in application interfaces that span multiple platforms? If cross-device consistency differs from the classic consistency, what insights can be transferred, what results are incompatible?
3. Limits: When does consistency support or inhibit inter-usability? While consistency is generally assumed to support transfer of skill between contexts, are there cases where consistency hinders more than it helps? Is consistency really supporting usability? When are devices too different so that transfer would be misleading? When does consistency prevent good design?
4. Evaluation: How can consistency be evaluated in the multi-device context and how can such measures integrate into the design process? Consistency today is implemented by means of the

implicit consistency of user interface toolkits, style guides, or by virtue of designer's expertise. Tools to support consistent design for cross-device applications have to be able to capture such aspects of consistency and to feed them back to the developer and designer. Methods to measure consistency and support consistent development will be collected and discussed. Further directions might be discussed and summarized.

Most of the position papers address questions 1-3, with a particular focus on question 2. Although there were few papers that discussed evaluations of consistency, we also know from the questionnaire that many participants are interested in discussing evaluation during the workshop. Points from the position papers for each of the questions are discussed in separate sections below.

A large number of the position papers express an interest in applying the idea of cross-platform consistency to some form of model-based design environment. The authors of these papers have many exciting ideas for merging these two areas, and we have included an extra section in this white paper to focus on this sub-topic.

It may be instructive for the workshop to consider applications for which cross-platform consistency is particularly important. The following applications were mentioned in the position papers:

- Multi-modality web applications, such as the shopping app in (Wiecha, Akolkar et al. 2006).
- Process Control, such as described on the third page of (Hajdukiewicz 2006)
- iTunes + iPod (Pyla, Tungare et al. 2006)
- Calendars (Pyla, Tungare et al. 2006)
- Home entertainment systems (Nichols, Myers et al. 2006; Trapp and Schmettow 2006)
- Others?

We may attempt to brainstorm a more complete list of applications during the workshop.

The next section of this paper discusses the definition of consistency based on previous research in the field. Following that section is a summary of the cross-platform consistency definitions that participants gave on our questionnaires. The next four sections summarize points from the position papers related to the four motivating questions that we asked in our workshop proposal. The following section discusses the relationship between model-based design and cross-platform consistency, as many of the position papers that we received had motivations in model-based work. We conclude with some generalizable principles gleaned from the position papers.

DEFINITIONS OF CONSISTENCY

To ground the discussion in some of the common references in the field of research on consistency in human-computer interaction, the most salient definitions of consistency will be summarized here. Nielsen, in the foreword to the "Coordinating User Interfaces for Consistency" book (Nielsen 1989) stated, "[…] consistency in computer systems constitutes a promise to the user. And it is not polite to break a promise." What exactly makes this politeness to the user?

First of all, consistency "has no meaning on its own; it is inherently a relational concept." (Kellogg 1989). Moreover, consistency can be considered as an agreement between two agents, e.g., the designer and the user, on how to interpret the user interface with respect to the task at hand (Reisner 1993).

"Consistency is assumed to enhance the user's possibility for transfer of skill from one system to another. By doing so, consistency leads to ease of learning and ease of use." (Nielsen 1989) Rule theoreticians like Anderson (Singley and Anderson 1989), and Kieras and Polson (Kieras and Polson 1985) explain this with the reuse of trained rules, saying that "Consistency facilitates positive transfer, speeding training, and improving retention of operation procedures." (Polson, Muncher et al. 1986).

As inherent part of the dialogue between designer an user, "Issues of consistency, both within and across applications and system environments, are inescapable for designers and users: users generalize, correctly or incorrectly, on the basis of their previous experience and what they know, and applications and environments are consistent or inconsistent in a variety of ways that impact use" (Kellogg 1989).

DEFINITIONS OF CROSS-PLATFORM CONSISTENCY

The workshop questionnaire asked each participant to give a personal definition for cross-platform consistency. These definitions can be broadly split into two categories: user-level and interface-level.

A typical user-level definition says that cross-platform consistency allows users to transfer knowledge to some degree between interfaces on different platforms. Several definitions specifically mention that the knowledge transfer should allow users to seamlessly change platforms while continuing to perform the same task. One user-level definition says consistency is "cognitive compatibility," or "the degree to which different platforms meet basic demands of human information processing and also the demands of highly trained user concepts." Another user-level definition pushes the idea of consistency to the emotional level, saying that "consistency provides the user with a feeling of being at home."

A typical interface-level definition says that consistent interfaces on different platforms will use the same design choices for the same functions, such as ensuring that the same widget always performs the same action. Various aspects of the interface are mentioned in these definitions, such as metaphor, content, structure, navigation, layout, look and feel, and interaction techniques. Most definitions mention only one or two aspects, although a few definitions explicitly prioritize several aspects of the user interface, such as one which states content > format/form > interaction. It is important to note that several definitions rate content as the most important aspect of the interface to be kept consistent.

All of the definitions can be read below in Appendix A.

QUESTION #1: DIMENSIONS

At least five different dimensions of cross-platform consistency are identified in the position papers: task, context, navigation, layout, and platform. These dimensions are discussed in detail below.

Task

According to several papers, task is perhaps the most important dimension (Ali 2006; Dittmar and Forbrig 2006; Hajdukiewicz 2006; Paterno and Mori 2006; Pyla, Tungare et al. 2006). Task is identified as one of three dimensions in (Trapp and Schmettow 2006). Task models may be strongly related to the navigation of a user interface, such as through the technique of attaching navigation operators to the non-leaf nodes of a task model (Ali 2006).

Task-action consistency (Dittmar and Forbrig 2006), which I believe is related to Norman's Gulf of Evaluation (Norman 1988), is noted as important aspect for multi-device systems. This is an interesting use of the word "consistency," because it implies a matching between the user's internal mental model and the interface rather than the normal notion of consistency being between two user interfaces.

Context

Contexts are a different reference frame through which to view consistency. In some cases contexts may be tasks, and there may be a complex stratification of contexts that are applicable to consistency. Contexts are used as the terminology here because each element of the application may belong to multiple contexts and those contexts may not be independent of each other. This differs from terminology such as dimensions, which implies independence between each dimension (Knapheide 2006). This idea of context may be relevant to some newer theories of language, which suggest that there are one or more reference systems with rigid rule sets. Breaking a rule allows for more flexibility in an utterance, but increases the probability that the utterance will not be understood by a second party. This seems to suggest an alternative goal, which would be for us to identify the "rules" for the design of an interface, and determine for each its relative flexibility/rigidity.

Navigation

Navigation is an important part of all user interfaces and it was referenced as a dimension of consistency in several position papers. Navigation was considered from several different perspectives. From a behavioral perspective, Ziefle et al. examined users' mental models of

navigation and showed that moving from hierarchically structured interfaces to other kinds of structures (particularly network structures) may prove to be particularly harmful to users (Ziefle, Arning et al. 2006). From a design perspective, Ali noted that "depending on the desired target platforms, different navigation styles might not just be desired – they might be a necessity" (Ali 2006). Wiecha et al. also noted that navigational structure can be shared between speech and graphical interfaces (Wiecha, Akolkar et al. 2006). Conveniently, the organization used in speech is often at the same level of detail as that needed for mobile devices with small screens. "Spatial order of available tasks" is listed a dimension by Trapp et al. (Trapp and Schmettow 2006), which is basically "the navigation to and selection of a specific task." In this paper however, spatiality is more in reference to interaction within a physical space, rather than panels on a user interface.

Layout

Luyten et al. (Luyten, Vermeulen et al. 2006) define a constraint-based algorithm for creating relative layouts where both the designer and the device can be taken into account during final layout. This approach allows for some amount of consistency between layouts between different devices. Layout has strong ties to navigation of course, but I felt that they differed enough to warrant the inclusion of layout as a separate dimension.

Platform (Look and Feel)

Several papers mentioned look-and-feel as a dimension (Alexander 2006; Trapp and Schmettow 2006). A particular challenge within this dimension is managing the conflict between native look-and-feel and cross-platform consistency. Alexander "believes the choice of where to follow native look-and-feel vs. where to have a consistent experience across platforms that may also express the corporate brand is a complex one which cannot be addressed with a simple overarching rule" (Alexander 2006). Some elements must stay consistent with platform look-and-feel, such as the positioning of OK and Cancel buttons, etc. Some elements can innovate in look but not feel, such as scrollbars and checkboxes (though scrollbars are often done wrong, while checkboxes are usually right). Collections of controls can often innovate on look-and-feel within platform, possible to be consistent with a corporate brand or perhaps the user's preference (Alexander 2006).

Relations Between Dimensions

Question #2 also identifies the relations between dimensions and their relative importance and an important are for discussion. The five dimensions identified here seem heavily inter-related, though there are some significant pairings that deserve extra discussion.

Task and context are closely related, particularly because contexts may be tasks in some cases. These two dimensions are also more abstract and do not focus on physical qualities of the interface unlike each of the other dimensions. These dimensions, at least for "good" interface designs, should also be the motivating factor behind most of the design decisions that lead to the physical design.

Navigation and layout are properties of the physical interface. Some people may consider these terms to be synonyms, but here they represent different aspects of the interface. Navigation is the process of finding a function, which may traverse multiple overlapping panels of a user interface. Layout refers to the placement of controls on a particular placement, which also greatly affects visual appearance. Navigation and layout are closely linked however, as interfaces with small screens may separate functions on to more overlapping panels and require more navigation. This also results in simpler layout for each particular panel.

Navigation and task are also related, though not as tightly as the two previous pairs. Ali showed that some navigation hints can be added to the task model (Ali 2006) to help with consistency.

Of all five dimensions, the least related to all of the others seems to be platform (look-and-feel). In general, the platform should be independent of the task and context dimensions. This is not always the case however, as some tasks may not be appropriate in some platforms (for example, watching television on a mobile phone does not necessarily make sense (Paterno and Mori 2006)). Platform and navigation are not entirely independent of each other either, although platforms must be substantially different in terms of screen size or modality before navigation is forced to differ significantly. Even when navigation is forced to differ, it may be the case that consistency is not affected unless the platforms do not allow for the same mental model of navigation (e.g. hierarchical

vs. network) (Ziefle, Arning et al. 2006). Layout is the most affected by platform, especially where platforms use different widgets or even different renderings of the same widget.

QUESTION #2: UNIQUE PROBLEMS

Ensuring cross-platform consistency requires the interface designer to address all of the challenges of creating a single user interface with the addition of extra dependencies between each of the target platforms. The problems identified in the position papers discuss some of the important cross-platform relationships that must be addressed.

User Interpretation and Behavior Shaping Constraints

Two of the position papers noted that the relation between form and function is not symmetrical (Hajdukiewicz 2006; Knapheide 2006). In other words, functions on different platforms of a cross-platform interface may work differently provided that the user always interprets the function to be the same. The form of the interface can also be thought of as a "behavior shaping constraint." "Forms that map to key behavior shaping constraints…should remain consistent or invariant; other aspects are less important in terms of consistency and are free to change based on the limitations of the platform" (Hajdukiewicz 2006). It seems then that a unique problem for cross-platform consistency is identifying forms that can change but always be interpreted to be similar by the user. Many of these issues exist in the design of platform widgets, which is explored in some depth in (Alexander 2006).

Another way to look at the interpretation problem is through a concept called "reference systems" (Knapheide 2006). This idea comes from a newer theory of language. When two people attempt to communicate, they draw from their own references systems to build their utterances and arguments. The rules that build these reference systems have varying levels of rigidity, and breaking a more rigid rule comes at a higher price than that of a more flexible rule. Breaking too many rigid rules may prevent any communication from taking place at all. If we take this analogy into our problem, then an alternate goal of the workshop might be to identify rigid rules that when broken create a source of confusion for users. These rules could then be used as our basis for judging whether an interface is consistent enough (i.e. the system is consistent enough if the user would not be confused). Are there common reference systems that we might easily apply to all interfaces, or are all these systems domain- or user-specific?

Seamless Task Migration, Continuity, Interaction Momentum

Seamless task migration (Pyla, Tungare et al. 2006) between interfaces on different platforms, alternately known as continuity (Denis and Karsenty 2003; Florins, Trevisan et al. 2004) or interaction momentum (Hajdukiewicz 2006), was a theme of several position papers. Interfaces that support this kind of feature might be termed continuous user interfaces (Pyla, Tungare et al. 2006). These ideas are unique to cross-platform applications, and their unique problems seem relevant to achieving cross-platform consistency.

One unique problem for enabling task migration is support for the transfer and recovery of state and activity context. When the user transitions to a different platform, state information from the initial platform must be transferred to the new platform. If this transfer of information is time-consuming, then the user must be made aware that this state transfer is happening in the background. This is one example where task migration may suggest avoiding consistency, at least initially, to ensure the user has a correct mental model of the system.

Another challenge for task migration is determining which tasks the user will want to migrate. Can all tasks be migrated across platforms, or must the designer carefully choose a subset?

Resource Issues

Other participants suggest that we should not "strive for seamless switches between devices," but rather "develop means for creating efficient task scenarios across devices" (Dittmar and Forbrig 2006). In other words, devices should support a subset of tasks that are reasonable for their platform. The task subset will be related to attributes such as screen size and other input/output capabilities. Even when task subsets are used to design a cross-platform interaction, users will want to re-use knowledge from other interfaces (Paterno and Mori 2006).

How do we identify the task subsets that make sense for a platform? It seems that these will be highly domain-dependent. Can this be generalized in any way?

High-level/Low-level Specification of Interfaces

A challenge for creating cross-platform interfaces is to create tools that allow specification of the interface at multiple levels (Wiecha, Akolkar et al. 2006). This allows the designer to specify an application's functionality at a high-level, which ensures some amount of consistency across different platforms (channels). Specification at lower-levels must also be supported however, so that designers can then rapidly refine the details of any one platform of an application without impacting the others. Another challenge is to create intelligent tools that can assess the impact on consistency of making low-level changes to a particular platform.

QUESTION #3: LIMITS

Grudin (Grudin 1989) gives many examples of places where consistency – narrowly labeled as constancy – leads to a less usable design than a seemingly inconsistent design. Several position papers identified specific cross-platform scenarios where consistency may not be the best choice. Most other papers seem to implicitly believe that consistency can be helpful by enabling such features as seamless task migration and avoidance of task disconnects (Pyla, Tungare et al. 2006).

It is also important to consider the generality of these scenarios. According to one position paper: "based on previous research, the aspects that should be consistent will be highly dependent on the application, domain, and context of use" (Hajdukiewicz 2006).

Perlman offers an alternate view: instead of ensuring consistency, we should attempt to avoid "unintentional inconsistency," which is defined "as the situation where two parts of a design differ for no good reason" (Perlman 2006). Unintentional inconsistency can be avoided in several ways. One primary means is to use a structured design process where the high-level design choices can be specified in one location. This allows changes to those decisions to be automatically incorporated across the system because all pieces of the system rely on the centralized design information to create their interfaces. Another method to avoid unintentional inconsistency is with measurement metrics. "A simple example is to measure, for each term (e.g., Search), how many times it appears in values compared to how many times a reference to it appears" (Perlman 2006). This approach is similar to the Sherlock system for testing consistency in desktop user interfaces (Mahajan and Shneiderman 1997).

Hindrances for Cross-Platform Consistency

Alexander describes many situations where the platform, corporate branding, and the user's mental model (consistency) may all conflict (Alexander 2006). For example, the open file dialog box is a native widget that must be used by applications for security reasons. Consistency may also be a limitation if it gives the impression that a device will have more data than it actually does, such as when synchronization of all data for an application is impractical or impossible (Pyla, Tungare et al. 2006).

Consistency may also conflict with good design if it dictates supporting a task that is not appropriate for the given device (Paterno and Mori 2006; Pyla, Tungare et al. 2006). For example, the iPod device and the iTunes application have different responsibilities for managing playlists. The iPod can choose playlists and play the music on a playlist, but it cannot be used to modify playlists. The editing feature is given only to the iTunes application. In this example, consistency would dictate that the iPod should also be able to edit playlists, but the Apple engineers did not include this feature because they felt the interface hardware could not easily support this function.

The form factor may also limit the usefulness of consistency. For example, a desktop calendaring application may default to showing a full month at a time, whereas it only makes sense for a smart phone to show the day view by default (Pyla, Tungare et al. 2006). Form factor may also require changes in structure and layout that affect consistency.

QUESTION #4: EVALUATION

Only one of the position papers that we received discussed behavioral studies of consistency (Ziefle, Arning et al. 2006), however the results from the questionnaires suggest that many of the participants are interested in evaluation. In this section I have also included some brief discussions of previous systems that have automated the evaluation of consistency for desktop applications (not cross-platform), which may be useful to include in our discussion.

Evaluation techniques for consistency break down into two categories, those that can be automated given some description of the user interface and those that require interaction from real users in a controlled environment.

Ziefle et al. examined the mental model of the user to determine whether the user had correctly interpreted the formatting of the interface and understand how misconceptions can affect performance (Ziefle, Arning et al. 2006). Their studies examined mental models for menu structure on a phone and PDA by asking users to perform a set of tasks and afterward to arrange a set of cards representing the steps in the menu according to their perceived structure. For the phone interface, which had a hierarchical structure, users with the correct mental representation solved 24.1% more tasks than those who did not. Results for the PDA also showed that users with adequate mental models of the structure also did better than those who did not.

GLEAN (Kieras, Wood et al. 1995) and Sherlock (Mahajan and Shneiderman 1997) provide automated evaluation of consistency for regular desktop interfaces. GLEAN requires models of the interface in the NGOMSL language, but can do analyses that test for knowledge transfer between different pieces of the interface or different interfaces. Sherlock operates on textual models of the interface written in its own language, and it provides translators to its language from standard Visual Basic formats. Sherlock uses a variety of metrics, including some of those mentioned by Perlman (Perlman 2006), to help the designer identify consistency problems in interfaces. The output of Sherlock can be difficult to interpret however, and it provides no assistance in actually making changes to the interface.

Richter (Richter 2006) suggests supporting users in the transition between platforms by facilitating the transformation of their mental model between designs. If a user knows a certain platform he should be able to easily transfer his knowledge, including knowledge about the spatial layout, by means of regular transformations. Based on this assumption a consistency measure is suggested that describes the regularity of a transformation for different properties of a user interface. This can be used for design-time support methods such as "consistency ghost" (Richter 2006).

CONSISTENCY AND MODEL-BASED DESIGN

A large portion of the position papers have some relation to model-based user interface development or model-driven engineering. It seems that many research systems have already included consistency as a factor or are considering adding some aspect of consistency.

Sottet et al. (Sottet, Calvary et al. 2006) describe a Model-Driven Engineering (MDE) framework for building applications. This process supports many libraries of transformations which take information in models and eventually produce a user interface. These interfaces have "plasticity," which might allow things like consistency to be added to the user interface. One idea is to automatically adapt the user interface to match the current "context-of-use," in which case consistency would be an important factor to take into account during adaptation.

Uniform is a new system for automatically generating personally consistent remote control interfaces based on knowledge of the user's previous experiences (Nichols, Myers et al. 2006). Uniform does not use an underlying task model, but approximates task consistency by ensuring the function are used in the way and located in the same place within the user interface. A knowledge base of functional similarity relations support a set of rules which manipulate the abstract user interface, resulting in a consistent final interface for the end user.

Supple uses optimization to automatically generate user interfaces, which allows it to naturally represent the cross-platform consistency as a tradeoff between optimal design for each platform and increased similarity of appearance between the two interfaces (Gajos, Wu et al. 2005).

The layout algorithm described by Luyten (Luyten, Vermeulen et al. 2006) and the task-navigation work by Ali (Ali 2006) are examples of new rules for model-based design environments that may improve the consistency of generated interfaces.

Many model-based environments rely on task models, but Dittmar et al. (Dittmar and Forbrig 2006) suggest that current task and dialogue models are not sufficiently detailed to model task-action consistency. In particular, the general task models used today are not sufficient to model the flow of task domain objects between applications. Representations that allow this will be important for model-based design of multi-device systems.

CONCLUSIONS

The goal of this workshop is to find answers to the four main questions stated above. We hope that through a discussion of these questions that we can build consensus of important areas for future work in cross-platform consistency and provide insight to each participant that can be integrated into their current research projects.

Another interest is to build a framework through which cross-platform consistency can be understood. A start may be to build on Hajdukiewicz's 4 principles for effective cross-platform design (Hajdukiewicz 2006):

1. Content that is critical to task-completion must be consistent in form across platforms
2. Secondary and supporting information do not need to be consistent (but can be if appropriate). Constraints of the platform should guide the design for this data
3. Navigation metaphors should be compatible [editor: consistent?] across platforms
4. Other aspects of application design should conform to effective user centered design practices for the platform considered.

A final goal of this workshop is work towards a future publication or other dissemination of the discussion and results from this workshop. The form that this publication will take has not yet been decided, and will be discussed in the final session of the workshop.

APPENDIX A. CONSISTENCY DEFINITIONS

The following are definitions of cross-platform consistency provided by the workshop participants.

- "We assume that users can apply different devices of an interactive system to accomplish their tasks. In this context, consistency ensures that a user can move between devices seamlessly with respect to a task. (Sub-)Tasks supported on different devices have to be mapped to the same abstract actions. An abstract action leads to the same abstract effect on all user interfaces."
- "A mental model of interaction, at some level – perhaps only conceptual, navigational, if not surface look & feel – that allows for prediction and transference of user skills from one platform to another."
- "Complete consistency would mean that if the user could only see the application portion of the UI, they would not even know what platform they were using – the behaviors and appearances would be identical. A more pragmatic approach is to find the right balance between optimizing for the platform and following its standards while still having the user be able to recognize the product as being the same (branding), knowing exactly how to use it from experience with it on another plaform, and most of all not being confused about which way to do things on this platform vs. another platform."
- "Minimally: Tognazzini 1990 – the same handle should always do the same action. Not the other way around."
- "As defined by human factors, consistency refers to the way same design choices (format and position) are applied in same contexts (platform is one aspect)."
- "Consistency is one means that provides the user with a feeling of being 'at home'."
- "For me consistency (invariance) across platforms is defined in multiple dimensions: content of what is conveyed across platforms; format/form of how this information is conveyed (e.g., are visual artifacts the same); interaction of how end users interact with the platform."

- “Part of the structure and of the user interface and navigation through the user interface for a set of supported tasks remains intact across platforms, while the user interface can have a different presentation.”
- “Consistency in cross-platform design involves making available a similar set of features on multiple devices (or, an appropriate subset of such features on devices with smaller form factor), keeping the look-and-feel and behavior of the application intact. The aim is to let users perform the same tasks on more than one platform through a UI that is almost similar on each platform.”
- “Consistency in cross-platform design describes the degree of similarity of the execution of a user task on different platforms. It relates to the platform look & feel as well as to the temporal and spatial order of subtasks.”
- “As psychologist, consistency is referring to what we call cognitive compatibility, thus the degree to which different platforms meet basic demands of human information processing and also the demands of highly trained user concepts which should be kept consistent across devices.”
- “A consistent view of a user’s information that scales across devices and displays. Interaction techniques might need to change but the UI metaphor stays consistent.”
- “The amount of (demonstrated) knowledge transfer that occurs across interfaces/platforms.”
- “a quality of interfaces for two or more devices in which the interface elements are similar enough to make the application interfaces of both devices feel consistent to the users.”
- “Consistency is a means to keep a particular design and interaction choices (as close as possible) across multiple platforms”
- “Consistency in user interfaces in general means that in similar situations the same design solution is adopted. This is true even in cross-platform design.”

REFERENCES

[1] Alexander, C. (2006). CHI Workshop Position Paper - Workshop on "The Many Faces of Consistency in Cross-platform Design". The Many Faces of Consistency in Cross-Platform Design Workshop at CHI'2006. Montreal, Canada.

[2] Ali, M. F. (2006). Navigation Consistency, or the Lack Thereof, in Cross-Platform User Interfaces. The Many Faces of Consistency in Cross-Platform Design Workshop at CHI'2006. Montreal, Canada.

[10] Denis, C. and L. Karsenty (2003). Inter-usability of multi-device systems: A conceptual framework. Multiple User Interfaces. A. Seffah and H. Javahery, John Wiley & Sons: 373-385.

[11] Dittmar, A. and P. Forbrig (2006). Task-Action Consistency in Multi-Device Systems. The Many Faces of Consistency in Cross-Platform Design Workshop at CHI'2006. Montreal, Canada.

[12] Florins, M., D. G. Trevisan, et al. (2004). The Continuity Property in Mixed Reality and Multiplatform Systems: A Comparative Study. CADUI'04, Funchal, Portugal.

[13] Gajos, K., A. Wu, et al. (2005). Cross-Device Consistency in Automatically Generated User Interfaces. Proceedings of the 2nd Workshop on Multi-User and Ubiquitous User Interfaces, San Diego.

[14] Grudin, J. (1989). "The Case Against User Interface Consistency." CACM 32(10): 1164-1173.

[15] Hajdukiewicz, J. (2006). Interaction Momentum - Industrial Application Design and Consistency Across Platforms. The Many Faces of Consistency in Cross-Platform Design Workshop at CHI'2006. Montreal, Canada.

[16] Kellogg, W. (1989). The dimensions of consistency. Coordinating User Interfaces for Consistency. J. Nielsen, Academic Press: 9-20.

[17] Kieras, D. and P. Polson (1985). "An Approach to the Formal Analysis of User Complexity." International Journal of Man-Machine Studies 22: 365-395.

[18] Kieras, D. E., S. D. Wood, et al. (1995). GLEAN: A Computer-Based Tool for Rapid GOMS Model Usability Evaluation of User Interface Designs. Eighth Annual Symposium on User Interface Software and Technology, Pittsburgh, PA.

[19] Knapheide, C. (2006). Which Consistency Are You Talking About? The Many Faces of Consistency in Cross-Platform Design Workshop at CHI'2006. Montreal, Canada.

[20] Luyten, K., J. Vermeulen, et al. (2006). Interfaces that can Shrink and Stretch: Multi-Device Plastic Layout Management. The Many Faces of Consistency in Cross-Platform Design Workshop at CHI'2006. Montreal, Canada.

[21] Mahajan, R. and B. Shneiderman (1997). "Visual and Textual Consistency Checking Tools for Graphical User Interfaces." IEEE Transactions on Software Engineering 23(11): 722-735.

[22] Nichols, J., B. A. Myers, et al. (2006). UNIFORM: Automatically Generating Consistent Remote Control User Interfaces. Proceedings of CHI'2006, Montreal, Canada.

[23] Nielsen, J. (1989). Coordinating User Interfaces for Consistency. San Francisco, CA, Morgan Kaufmann.

[24] Norman, D. A. (1988). The Design of Everyday Things.

[25] Paterno, F. and G. Mori (2006). Interaction Semantics Continuity in Cross-Platform and -Modality Interface Design. The Many Faces of Consistency in Cross-Platform Design Workshop at CHI'2006. Montreal, Canada.

[26] Perlman, G. (2006). Avoiding Unintentional Inconsistency. The Many Faces of

[27] Consistency in Cross-Platform Design Workshop at CHI'2006. Montreal, Canada.

[28] Polson, P. G., E. Muncher, et al. (1986). A test of a common elements theory of transfer. SIGCHI conference on Human factors in computing systems, New York, NY, ACM Press.

[29] Pyla, P. S., M. Tungare, et al. (2006). Multiple User Interfaces: Why Consistency is Not Everything, and Seamless Task Migration is Key. The Many Faces of Consistency in Cross-Platform Design Workshop at CHI'2006. Montreal, Canada.

[30] Reisner, P. (1993). "APT: A description of user interface inconsistency." International Journal of Man-Machine Studies 39: 215-236.

[31] Richter, K. (2006). Transformational Consistency. Computer-Aided Design of User Interfaces (CADUI), Bucharest, Romania.

[32] Singley, M. K. and J. R. Anderson (1989). The transfer of cognitive skill. Cambridge, MA, Harvard University Press.

[33] Sottet, J.-S., G. Calvary, et al. (2006). Towards Mapping and Model Transformation for Consistency of Plastic User Interfaces. The Many Faces of Consistency in Cross-Platform Design Workshop at CHI'2006. Montreal, Canada.

[34] Trapp, M. and M. Schmettow (2006). Consistency in use through Model based User Interface Development. The Many Faces of Consistency in Cross-Platform Design Workshop at CHI'2006. Montreal, Canada.

[35] Wiecha, C., R. Akolkar, et al. (2006). Design methods and software tools for consistency in multi-channel web applications. The Many Faces of Consistency in Cross-Platform Design Workshop at CHI'2006. Montreal, Canada.

[36] Ziefle, M., K. Arning, et al. (2006). Cross platform consistency and cognitive compatibility: the importance of users' mental model for the interaction with mobile devices. The Many Faces of Consistency in Cross-Platform Design Workshop at CHI'2006. Montreal, Canada.