2017 IEEE International Conference on Systems, Man, and Cybernetics (SMC)
Banff Center, Banff, Canada, October 5-8, 2017

An Open Source Adaptive User Interface for Network Monitoring

Sean W. Kortschot, Dusan Sovilj, Harold Soh, Greg A. Jamieson, Scott Sanner, Chelsea Carrasco
Mechanical and Industrial Engineering
University of Toronto
Toronto, Canada

Scott Ralph & Scott Langevin
Uncharted Software
Toronto, Canada
sralph@uncharted.software,
slangevin@uncharted.software

Abstract—**Decision support systems for network security represent a critical element in the safe operation of computer networks. Unfortunately, due to their complexity, it can be difficult to implement and empirically assess novel techniques for displaying networks. This paper details an open source adaptive user interface that hopes to fill this gap. This system supports agile development and offers a wide latitude for human factors and machine learning design modifications. The intent of this system is to serve as an experimental testbed for determining the efficacy of different human factors and machine learning initiatives on operator performance in network monitoring.**

Keywords—adaptive user interface; active network monitoring; experimental platform; open-source

I. Introduction

Cyber security is becoming increasingly important as a growing number of organizational activities are moving online [1]. With the additional reliance on internet-based computational activities, there has also been a surge in cyber-attacks [2]. As such, a strong focus on cyber security is necessary for the safe and successful operation of most organizations [3]. Although the infrastructure to prevent attacks from being successful is integral to effective cyber-security, there are still attacks that circumvent these safeguards [4]. When attacks do bypass security measures, active network monitoring (ANM) is necessary.

ANM is the process of detecting and mitigating the effects of network intrusions or attacks, either in real time or forensically [5]. Since operators are typically required to observe networks composed of massive amounts of highly distributed data, a critical aspect of the ANM task involves parsing through irrelevant data to find the information that is pertinent to the current system state [6]. This task becomes especially challenging since the pertinent information is constantly changing. ANM operators therefore rely heavily on decision support systems (DSSs) to aid their navigation through the network.

DSSs typically operate by comparing known attacks against ongoing activity in the network to provide a ranked list of areas that the operators should attend to [6]. Operators then combine this information with their tacit knowledge of the network and direct their attention towards the area that they perceive to be of the most relevance [7]. Advances in machine learning and data mining have allowed for more effective DSSs to be developed [8]. However, since novel attack methodologies are constantly being invented to counteract these security measures [9], designers must develop robust solutions that are capable of adapting to unanticipated future attacks. This necessitates a strong focus on the human factors issues related to cyber security and DSSs, so that when these unknown attacks happen, the human operator is in the best position to address them.

One of the ways that human factors are being considered in the design of future DSS systems is through adaptive user interfaces (AUIs). AUIs modify the content of an interface in response to changing system and user states. Since ANM typically involves such massive amounts of data, AUIs can be especially useful, helping to adjust the information content to that which is most likely to be useful to an operator in a given scenario. Effective AUIs necessitate the successful bridging of machine learning and reasoning methods with human factors considerations.

Unfortunately, DSSs are typically complex, robust systems, requiring a great deal of back- and front-end development. Therefore, gathering empirical evidence of the efficacies of various design methodologies can be challenging. Because of this, a lower-fidelity platform that is capable of imposing the same types of attentional demands that real-world operators face is necessary. This can provide empirical evidence for novel features and ensure the proactive development of defense solutions.

This paper will discuss a novel experimental platform that is capable of supporting a variety of machine learning applications and adapting the interface to those learning processes. Furthermore, we will describe how evidence can be gathered to support both human factors and machine learning initiatives and how this evidence can be incorporated in future designs. Our sole focus is describing our platform, and therefore we will not discuss efforts to improve machine learning recommendations.

II. Related Work

In the cyber-security domain, there have been plenty of solutions providing operators with a tool to address their needs

– detecting threats and investigating network alerts. Ocelot [10] introduces a multi-faceted graph visualization with additional features: node grouping and group assessment, communication exploration, quarantine labeling, and a designated panel for alerts. The main development of Ocelot is its multi-view take on networks to facilitate different viewing perspectives to the operators.

Attack-graphs [9], [11] have been another way of presenting possible attack sequences within networks that are subject to investigation. The PERCIVAL [12] system provides the operator with a visual list of detected threat paths based on network statistics alongside a complete network display with a component to assess the effectiveness of the mitigation actions.

Although the effectiveness of the above mentioned solutions have been demonstrated, they still do not consider adaptations as part of the interface, and instead focus on providing specific network details in separate panels besides the graph/network window to facilitate design goals.

Adaptive visualization represents a potential way to improve visualizations by modifying visual elements in the interface in response to changing system and user states. Adaptations to the interface have been designed to address information retrieval [12], [13], various analytic-task solving problems [14], and displaying large-scale social networks [15]. However, such adaptations have not been investigated within visualizations developed for cyber-security related tasks.

In this paper, we are bridging this gap by offering a platform where communication networks can be visually investigated for potential threats while visual elements are adapted through a learning model. This model is updated in an online fashion incorporating user feedback as part of the learning process.

III. Design Requirements

We had four primary classes of design requirement leading to the development of the platform. First, the platform had to adequately simulate the psychological and task demands imposed on operators by the ANM task without necessarily replicating it. The second and third requirements were that the platform allow for manageable integration of both machine learning and reasoning research initiatives, as well as human factors research initiatives. Finally, the experimental platform needed to allow for easy and reliable experimentation. Each of these objectives had several sub-goals, which will be discussed in turn below and are presented in Table 1.

A. Simulating Active Network Monitoring

As stated earlier, the experimental platform did not need to be a replication of a full scale DSS. Instead, it simply needed to adequately elicit some of the cognitive demands that can hinder performance in this type of environment - a large heterogeneous communications network. Moreover, due to a scarcity of network operators, the majority of experimentation using a platform like this will be done on either trainees or completely novice operators. Therefore, it is actually preferred to have a stripped down DSS, as it is more approachable for a novice.

The core requirements that we identified were 1) the ability to host a sufficiently large network to necessitate zooming and panning behaviour through a network map, 2) the ability to inspect communication contents, 3) the ability to tag those communications as either normal, malicious, or suspicious, and 4) the ability to support a recommendation list, as this is a cornerstone of most real-world DSSs [11].

B. Machine Learning Initiatives

To accommodate a machine learning DSS, a learning model should provide three key features: feasible ranking of communications, computational efficiency, and ability to learn in highly imbalanced data.

To provide the user with a meaningful and useful recommendation list, the machine learning model should provide a real-valued score indicating level of maliciousness rather than a binary malicious/normal answer (or a multi-label outcome). Certain models have an implicit way of providing this score in terms of probability values (statistical models), while others provide this value through some form of distance-based metric (clustering algorithms). Given the operator input for tagged communication, the model should be able to account for this newly received content-class pair, and update the rankings for all other communications in the network. This condition requires a highly efficient model in terms of computational time, which involves combining the retraining (or online updating) and reranking phases. Ideally, processing time should be minimized, as this can improve visual momentum [16].

In real-world communication networks, the vast majority of communication patterns are deemed as normal with only a fraction of all patterns being worthy of investigation. As such, there is a massive difference in available samples for class types leading to unbalanced data sets for training purposes. This situation has to be accounted for during the modelling phase in order to yield reliable scores during platform runtime.

Given the amount of data associated with communication networks, it is possible to have several operators, or experts, responsible for smaller subdomains of the network or specific tasks within the network. From a machine learning perspective, each expert can be represented as a unique model that would be designed to tackle that particular task. During runtime, the goal is to choose a model that would provide the user with the most relevant recommendation list for the user's current task. In other words, we wanted to match user's current task with the task that was most similar to one of the experts. This can be done in an automatic manner depending on scores returned by each model, making this approach highly modular in a sense that different models can be used for different tasks if necessary. This approach can be seen as a mixture of experts modelling or collaborative filtering, depending on how the training phase is set up to exploit knowledge of experts.

TABLE 1. SUMMARY TABLE OF DESIGN INITIATIVES WHEN DEVELOPING THE EXPERIMENTAL PLATFORM.

Initiative	Goal	Details
Simulate the demands of the ANM task	Host a communications network	-Needs to be large enough to elicit information overload -Represent a multitude of comms. networks (e.g., email, packets, social)
	Allow for inspection of communications (comms.)	-Inspect content -Inspect Sender → Receiver -Directionality of comms.
	Allow for tagging of comms.	-Mark as malicious, suspicious, or normal
	Recommendation list	-Recommend areas to attend to
Machine Learning and Reasoning research initiatives	Recommendation list	-Associated score that can adapt to user-input
	Computational efficiency	-Fast learning model allowing seamless integration with rest of interface
	Robust learning	-Learning in low-sample scenario and unbalanced data
	Exploiting domain knowledge	-Reusing pretrained models acting as several domain experts
	Aggregation of domain knowledge	-Combining multiple learners with a modal framework
Human Factors research initiatives	Easily modifiable stylesheet	-Representation attributes (e.g., colours, weights, shapes, etc.)
	Robust logging system	-Track user interactions to allow for psychological state inferences
	Style sheet and machine learning and reasoning/logging system integration	-Stylesheet needed to be integrated with both recommendations and logging system for adaptive visualizations
Experimental	Multiple experimental conditions	-Easily host and shift between experimental conditions
	Multiple experimental tasks	-Easily host and shift between different experimental tasks
	Robust logging system	-Track all user interactions with the system at fractions of a second intervals

The platform also provides an option to choose between two modelling approaches. The first uses a single pretrained model, a global ranker, that is updated during execution time. This model learns to distinguish between all available classes that represent both normal and malicious content. The second approach uses collaborative filtering, which involves several models where each expert is trained on a specific set of data samples representing one malicious class (out of several). During runtime, this pool of expert models is expanded with a 'user' model that is constantly updated as newly labelled samples become available while expert models are kept intact. Recommendation scores in this approach are weighted averages across all models where larger weights indicate higher confidence that a model is appropriate for the contents the user is examining.

The basic building block for machine learning initiative is the logistic regression which is a fast non-linear model that can handle conflicting data samples. In the case of collaborative filtering, expert models plus the user model are in fact logistic regression models each trained on a separate data set.

C. Human Factors Initiatives

The human factors requirements are heavily tied to the requirements of the other elements of the system. On an elementary level, having an easily modifiable stylesheet is necessary to control various graphical elements in the user interface. However, the bulk of human factors considerations within our current research program revolve around the adaptive interface elements of the platform. Therefore, our stylesheet needed to be not only modifiable, but also connected to the machine learning and reasoning elements of the system so that the styles could change as the recommendation system learned from the user's input. Furthermore, we needed the stylesheet to be integrated with the logging system, as we aim to implement adaptations based not only on the learning systems' recommendation algorithms, but also based on how the user is interacting with the system.

D. Experimental Initiatives

From an experimental standpoint, there were three critical elements in the platform. The first two were that it needed to be capable of supporting multiple experimental tasks and conditions. The tasks in our first experiment involved different worms being hidden in the network with an infectious spreading mechanism. This represents just one of many possible scenarios that can be easily designed for, using the experimental platform. The experimental conditions in our first study were two different types of machine learning algorithm that drove the recommendation list.

The platform was designed with a modular perspective in mind to enable further experimentation with different learning models driving the recommendation list. The final experimental requirement was that there needed to be a robust logging system that was capable of recording every interaction that the participant executed during experimental trials. Furthermore, the state of the display needed to be captured as well. This meant that all visible nodes in the network map and all visible recommendations needed to be captured for every action that was recorded. Moreover, the zoom levels and coordinates of the viewport were needed to understand how the user was navigating throughout the network. Essentially, by recording all of these attributes we were able to recreate scenarios following the experimental trials and perform detailed statistical analysis on how users' behaviour changed as a function of system state.

IV. DESIGN SOLUTION

Our final design solution to the above listed requirements was a two-paneled interface consisting of a network map, which occupies approximately two thirds of screen real estate, and a details panel, which occupies the remaining area (see

Figure 1). The network map takes a fairly straightforward approach [15] where the sender and receiver are represented by circular nodes. The communication (in this example, emails) is represented by a square node, which is connected to the sender and receiver nodes via edges, indicating the directionality of the communication. We set the email nodes to be invisible to the user, which gave the appearance that emails were represented by edges.

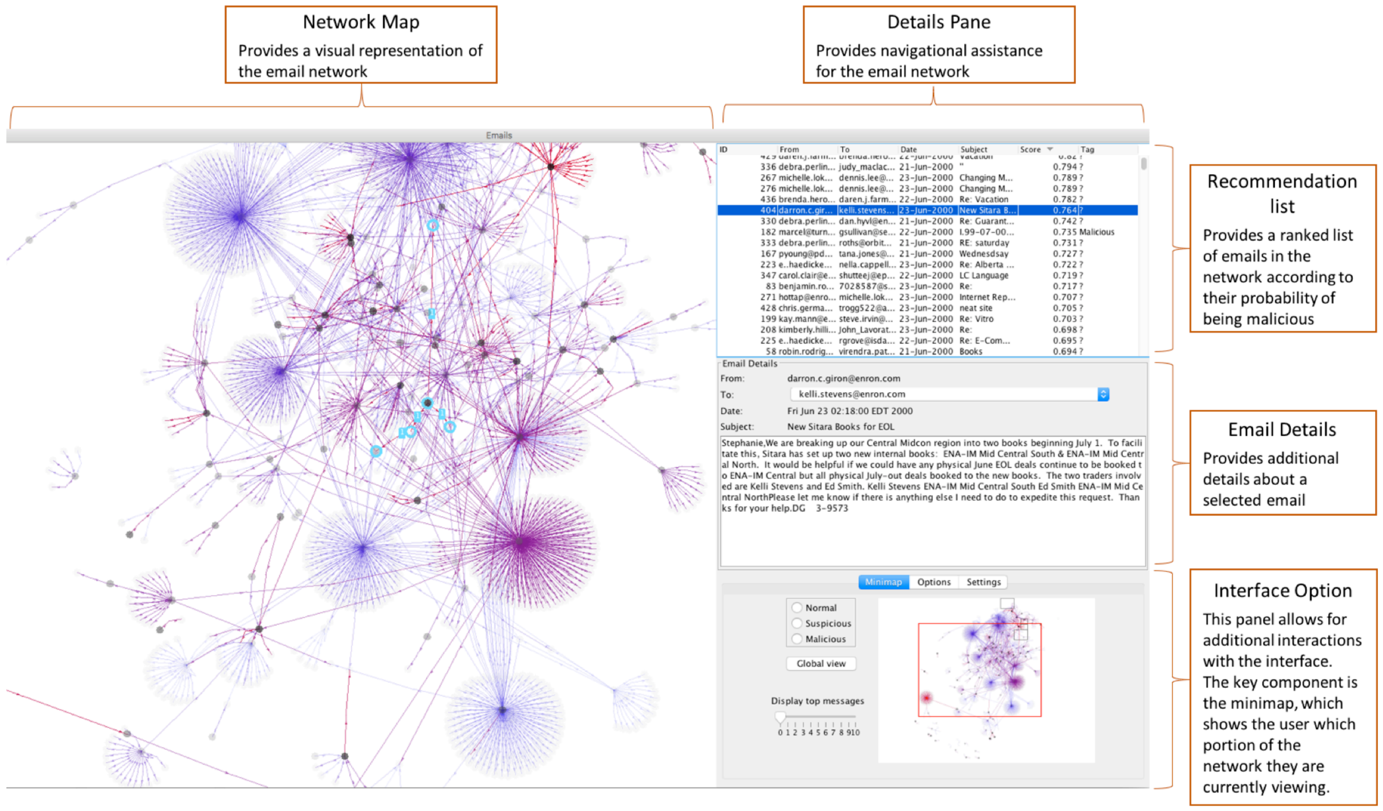
The details panel consists of three main elements; an adaptive recommendation list, an email details pane, and an interface options and navigation pane. The recommendation list provides a ranking of every communication in the network and is adjusted based on user input and the learning algorithm – global or collaborative approach. The email details pane provides available information about a selected email, either in the map itself or the recommendation list.

The interface options pane contains several different features. The first, and most significant, is a minimap of the network. This tool allows the user to see where their current viewport lies within the larger network. Users can also use the minimap to navigate to new points in the network by clicking that point in the map. This feature becomes especially useful when networks are larger. On the same panel as the minimap, the user has the option to tag an inspected email as malicious, suspicious, or normal. Again, the current iteration of the recommendation list will learn from this behaviour according to a global model, teaching the recommender what characteristics to look for in each of the three choices. The colours of the edges (representing individual emails in this case) change from blue to red in accordance with the score such that the highest scored emails are the most red. By having the tagging options on the same page as the minimap, the user is able to see how their tag influences the network as a whole via changes in colour. They can then use these changes to determine if there are new areas in the network that they should navigate to. These options are all easily modifiable and can be adjusted in accordance with specific experimental requirements.

The interface options pane also allows for filtering options to be integrated. We have built in a date-based filtering option that limits the network map to emails that were sent within a specified range. However, alternate filtering options are feasible with only minor adjustments. The filtering option also has a search function, which will display only emails containing the search terms. Finally, there is an options tab, which allows for any remaining options to be included by the experimenter. These options typically involve experimental conditions and tasks, but can also include display features and stylesheets.

A. Platform Capabilities

The above mentioned design requirements have enabled the fulfillment of three main platform capabilities, which allow for the expansion of experimental initiatives. First, the platform is capable of supporting any type of communications network. Therefore, future experiments can evaluate a wide array of work domains. Second, the recommendation system (i.e., list and suggested areas) is capable of supporting broad range of learning models. Again, this allows for future researchers to


Network Map
Provides a visual representation of the email network
Details Pane
Provides navigational assistance for the email network
Recommendation list
Provides a ranked list of emails in the network according to their probability of being malicious
Email Details
Provides additional details about a selected email
Interface Option
This panel allows for additional interactions with the interface. The key component is the minimap, which shows the user which portion of the network they are currently viewing.
Emails
Minimap
Options
Settings
Normal
Suspicious
Malicious
Global view
Display top messages
0 1 2 3 4 5 6 7 8 9 10

ID	From	To	Date	Subject	Score	Tag
336	debra.perlin...	judy_maclac...	21-Jun-2000	"	0.794	?
267	michelle.lok...	dennis.lee@...	23-Jun-2000	Changing M...	0.789	?
276	michelle.lok...	dennis.lee@...	23-Jun-2000	Changing M...	0.789	?
436	brenda.hero...	daren.j.farm...	22-Jun-2000	Re: Vacation	0.782	?
404	darron.c.gir...	kelli.stevens...	23-Jun-2000	New Sitara B...	0.764	?
330	debra.perlin...	dan.hyvl@en...	21-Jun-2000	Re: Guarant...	0.742	?
182	marcel@turn...	gsullivan@se...	22-Jun-2000	I.99-07-00...	0.735	Malicious
333	debra.perlin...	roths@orbit...	21-Jun-2000	RE: saturday	0.731	?
167	pyoung@pd...	tana.jones@...	21-Jun-2000	Wednesdsay	0.727	?
223	e..haedicke...	nella.cappell...	23-Jun-2000	Re: Alberta ...	0.722	?
347	carol.clair@e...	shutteej@ep...	22-Jun-2000	LC Language	0.719	?
83	benjamin.ro...	7028587@s...	23-Jun-2000	Re:	0.717	?
271	hottap@enro...	michelle.lok...	23-Jun-2000	Internet Rep...	0.707	?
428	chris.germa...	trogg522@a...	23-Jun-2000	neat site	0.705	?
199	kay.mann@e...	steve.irvin@...	23-Jun-2000	Re: Vitro	0.703	?
208	kimberly.hilli...	John_Lavorat...	23-Jun-2000	Re:	0.698	?
225	e..haedicke...	rgrove@isda...	22-Jun-2000	Re: E-Com...	0.695	?
58	robin.rodrig...	virendra.pat...	21-Jun-2000	Books	0.694	?

Email Details

From:	darron.c.giron@enron.com
To:	kelli.stevens@enron.com
Date:	Fri Jun 23 02:18:00 EDT 2000
Subject:	New Sitara Books for EOL

```
Stephanie,We are breaking up our Central Midcon region into two books beginning July 1.  To facili
tate this, Sitara has set up two new internal books:  ENA-IM Mid Central South & ENA-IM Mid Centr
al North.  It would be helpful if we could have any physical June EOL deals continue to be booked t
o ENA-IM Central but all physical July-out deals booked to the new books.  The two traders involv
ed are Kelli Stevens and Ed Smith. Kelli Stevens ENA-IM Mid Central South Ed Smith ENA-IM Mid Ce
ntral NorthPlease let me know if there is anything else I need to do to expedite this request.  Than
ks for your help.DG    3-9573
```

Fig. 1. The standard layout and feature-set of the experimental platform.

examine the relative efficacies of different models. Finally, the platform is capable of supporting a multitude of human factors design considerations and methodologies with minimal obtrusion into other areas of the system. These design considerations come on the front-end of the system in terms of stylesheet and general visualization modifications, and also on the back-end of the system by integrating human factors considerations directly into the recommendation system.

For access to the source code, please contact the author at sean.kortschot@mail.utoronto.ca.

V. Future Work

Although the current version of the experimental platform is relatively robust, there are a variety of areas of future work to consider. The first, and perhaps the most significant, is developing the capability to support dynamic networks. This facility will require significant development work and represents the biggest hurdle moving forward. We also hope to increase the scalability of the system to support even larger, more representative networks. However, because this platform is open-source, other researchers can explore further venues of extension.

References

[1] R. K. Goutam, “Importance of Cyber Security,” *Int. J. Comput. Appl.*, vol. 111, no. 7, pp. 14–18, 2015.

[2] N. Ben-Asher and C. Gonzalez, “Effects of cyber security knowledge on attack detection,” *Comput. Human Behav.*, vol. 48, pp. 51–61, 2015.

[3] B. Schneier, “The future of incident response,” *IEEE Secur. Priv.*, vol. 12, no. 5, pp. 96–97, 2014.

[4] A. Ahmad, J. Hadgkiss, and A. B. Ruighaver, “Incident response teams - Challenges in supporting the organisational security function,” *Comput. Secur.*, vol. 31, no. 5, pp. 643–652, 2012.

[5] A. D’Amico, K. Whitley, D. Tesone, B. O’Brien, and E. Roth, “Achieving Cyber Defense Situational Awareness: A Cognitive Task Analysis of Information Assurance Analysts,” *Proc. Hum. Factors Ergon. Soc. Annu. Meet.*, vol. 49, pp. 229–233, 2005.

[6] J. R. Goodall, W. G. Lutters, and A. Komlodi, “Developing expertise for network intrusion detection,” *Inf. Technol. People*, vol. 22, no. 2, pp. 92–108, 2009.

[7] N. A. Alrajeh, S. Khan, and B. Shams, “Intrusion Detection Systems in Wireless Sensor Networks: A Review,” *Int. J. Distrib. Sens. Networks*, vol. 2013, pp. 1–7, 2013.

[8] A. L. Buczak and E. Guven, “A survey of data mining and machine learning methods for cyber security intrusion detection,” *IEEE Commun. Surv. Tutorials*, vol. 18, no. 2, pp. 1153–1176, 2015.

[9] K. Sharma, A. Singh, and V. P. Sharma, “SMES and Cybersecurity Threats in E-Commerce,” *EDPACS*, vol. 5–6, pp. 1–49, 2009.

[10] D. L. Arendt, R. Burtner, D. M. Best, N. D. Bos, J. R. Gersh, C. D. Piatko, and C. L. Paul, “Ocelot: User-centered design of a decision support visualization for network quarantine,” *2015 IEEE Symp. Vis. Cyber Secur. VizSec 2015*, pp. 0–7, 2015.

[11] C. C. Gray, P. D. Ritsos, and J. C. Roberts, “Contextual network navigation to provide situational awareness for network administrators,” in *Visualization for Cyber Security*, 2015.

[12] M. Angelini, N. Prigent, and G. Santucci, “PERCIVAL: proactive and reactive attack and response assessment for cyber incidents using visual analytics,” in *Visualization for Cyber Security*, 2015.

[13] L. Williams, R. Lippmann, and K. Ingols, “GARNET: A Graphical Attack Graph and Reachability Network Evaluation Tool,” in *Visualization for Computer Security. Lecture notes in computer science*, J. R. Goodall, G. Conti, and K. L. Ma, Eds. Berlin, Hedelberg: Springer, 2008.

[14] J.-W. Ahn and P. Brusilovsky, “Adaptive visualization for exploratory information retrieval,” *Inf. Process. Manag.*, vol. 49, no. 5, pp. 1139–1164, 2013.

[15] L. Shi, N. Cao, S. Lio, W. Qian, L. Tan, G. Wang, J. Sun, and C.-Y. Lin, “HiMap: Adaptive visualization of large-scale online social networks,” *Vis. Symp.*, 2009.

[16] K. Nazemi, C. Stab, and A. Kuijper, “A reference model for adaptive visualization systems,” *Proc. 14th Int. Conf. Human-Computer Interact. Des. Dev. approaches*, pp. 480–489, 2011.