# MultiStates: Monitoring Databases With Acoustic and Intuitive Perspective Wall Interaction

**Thomas Gladisch**
Department of Computer Science
Dresden University of Technology
Dresden, 01062 Germany
Thomas.Gladisch@inf.tu-dresden.de

**Dirk Willuhn**
IBM Deutschland Research & Development GmbH
Boeblingen, 71032 Germany
+49 7031 164814
dwilluhn@de.ibm.com

**Gerhard Weber**
Dept. of Computer Science
Dresden University of Technology
Dresden, 01062 Germany
+49 351 46338467
Gerhard.Weber@inf.tu-dresden.de

**ABSTRACT**

In this paper, we describe the prototypical implementation and evaluation of a database performance monitoring tool for large database management systems (DBMS). These DBMS provide the technological background for many complex e-Government applications and the availability of the managed data is crucial. The implementation focuses on the creation of application-specific gestures on a touch input device, such as the Apple iPhone 3G with software version 2.2.1. A perspective wall is used to display the data in conjunction with an acoustic indicator for navigating through the information space non-visually. We report on an exploratory investigation of the prototypical monitoring tool based on an evaluation with two groups of users: inexperienced users with no database-related professional background and users, whose daily work is closely related to database monitoring. We conclude, knowledge in the area of the respective application is helpful to make better use of the prototypical tool. Furthermore, the flexibility of the perspective wall as the visualization of choice is shown by the good overall user acceptance. Finally, the acoustic indicator gives an idea of how to support even visually impaired users in finding occurrences of problems in large information spaces, such as database performance criteria.

**Keywords**

Mobile interfaces, gestures, visualization

## INTRODUCTION

### The increasing growth of digital data

Since the number of people using a computer and surfing the Internet grew from about 19.5 million in 1997 [1] to 1.2 billion [2] today, it is obvious that the amount of data stored and processed grew accordingly. It is expected that the total amount of data stored in 2006 (281 Exabyte) will be surpassed tenfold in 2011 without an end of increase in sight [3]. However, not only the amount of data increases, but also the availability of relationships between the data. As it became popular with the term "Web 2.0" the so-called "semantic web" tries to relate isolated pieces of information to each other, in order to create well-structured and accessible information. All of this data needs to be organized, stored and made accessible for users, whether these are practitioners, developers or even end-users. The need for systems taking care of the data management is reflected in the growing number of database management system in the public sector.

### DBMS in e-Government applications

In order to provide centralized data management and to use synergetic effects of shared knowledge domains, many regional and national governments in Germany enforce the establishment of new IT infrastructures. Examples are the LUSD system (Lehrer und Schüler Datenbank, teacher and pupil database [4]) in Hesse, Germany as well as the German Patent and Trademark Office (Deutsches Patent- und Markenamt [5]). The first one has been built to create a central access point for teachers and state officials to get an overview of all registered pupils in Hesse and to provide a communication platform for all persons related to teaching in schools. Thus, the system consists of very sensible data, which is important to ensure the organizational structure of the Hessian schools. The latter system offers overview, search and registration of patents and trademarks to end-users. It contains all patents and trademarks registered in Germany and therefore relies on a large and complex information set.

### The importance of mobile database monitoring

As the description of the two examples above suggests, the availability of the maintained data is crucial. Therefore, database administrators (DBAs) take care of monitoring and optimizing the databases, keeping the systems up and running. This task needs to be performed throughout the day, independently from the current position of the DBA. A failure of a database might not only result in a costly unavailability of data, but also in a loss of sensible data, which is inacceptable especially for e-Government applications. Especially the complexity of current solutions for database performance monitoring implies the need for a desktop computer system or at least a fully featured web browser. This contradicts the prerequisites mentioned before and leads to the idea of creating a mobile application with a reduced feature set, similar to the "schema later" approach by Jagadish *et al.* [7], which intends to hide the complexity of a system from the user. As a result, good interfaces for such complex tasks need to be simple enough to make them understandable even for people without knowledge of the whole system. This can be achieved by providing a certain level of "ad hoc"-ness [6] when working with the application, which is similar to the ease of input in search engines, where a simple text field is sufficient. Combined with less textual and more graphical information and a reduced informational depth, this is a promising approach for creating a novel interface for mobile database performance monitoring.

## RELATED WORK

As mentioned previously throughout this work, many similar approaches in terms of visualization and support for visually impaired people are available. Nevertheless, they have mostly a more narrow focus of application. An example for an application specialized on supporting blind users in using touch-based interfaces is Slide Rule [15], where the graphical user interface (GUI) is replaced by an overlay, which recognizes new application specific gestures and relates them to content laid out in a grid. The earPod [19] application also leverages audible feedback in combination with a click wheel touch input.

Similar to the tilt control of MultiStates is the speed-dependent automatic zoom approach of Eslambolchilar *et al.* [20] where the tilt angle influences the degree of magnification (which is not supported in MultiStates), as well as the scroll direction and scroll speed. Since the amount of available visualizations in products and prototypes only the two most influencing publications are mentioned here: At first, the perspective wall concept by Mackinlay *et al.* [21] and second, the ZuiScat [13] system, which incorporates a useful combination of geometric and semantic zoom, which has been used in the MultiStates variant of the perspective wall. Although the main aim was to make a complex database monitoring system usable on a mobile device by providing intuitive interaction and helpful visualization while supporting individualization, we discovered ideas for building a foundation to support blind and visually impaired users.

## DESIGN OF MULTISTATES

To offer a mobile solution, we designed the MultiStates prototype, which makes use of geometric and semantic Zoom, leveraging a modified version of the perspective wall [26]. MultiStates runs on an Apple iPhone 3G, currently with software version 2.2.1. MultiStates provides a new way of monitoring databases on the go by combining two IBM products used as desktop solutions: IBM DB2 Performance Expert V3 [8] and IBM Data Studio Administration Console [9]. The focus lies on the creation of the so-called "Health Summary" in conjunction with the dashboard view of the latter product. The prototype uses dummy data, which are not synched with a server, representing a use case to discover and analyze lock conflicts and deadlocks.

### Applied interface design principles

When it comes to working on a mobile platform, users need to be supported by a tailored interface, which comprises intuitive forms of interaction and simple visual cues. Furthermore, the option to configure or filter the displayed data should be given for expert users. We used Nielsen's [10] user interface design "rules of thumb" as a checklist for ensuring a flawless transition from a stationary to a mobile interface. The key attributes for MultiStates are: user control and flexibility, as well as consistency and visual feedback. User control and flexibility imply that only the user initiates interaction and that he/she may chose from a set of input method his/her favorite. Furthermore, the option to reset the view of the application is important to allow the user to go back to his/her starting point. Consistency and feedback are provided through the consistency with the original desktop product, by using e.g. the same vocabulary and color-coded information.

These four criteria mentioned above also reduce the memory load for the end user, as well as support recognition of items and interactions rather than enforcing the need to recall complex interfaces and interaction techniques. How this has been achieved is described in the following subsections.

### Database state visualization

To ensure a clean and simple visualization on the screen of the iPhone, the perspective wall has been chosen. This selection results from a comparison of multiple visualization techniques, which concentrate on the presentation of a central focus region, while avoiding the desert fog [11] problem, where users get lost in their potentially large dataset. Although many systems, such as DateLens [12] or ZuiScat [13] support geometric and semantic zoom to reduce screen clutter, they do not provide a fluent transition between focus and context. Hence, the perspective wall was used to visualize the Health Summary. Although the perspective wall seems mainly suitable for showing information related to time, by using time as the measure for a long horizontal X-axis when scrolling, it is feasible to use it for database states.

In order to make the perspective wall usable for our database monitoring purposes, the dataset needed to be matched to this visualization. Since the displayed data needs to be aligned along a small Y-axis and a longer X-axis, we used the performance criteria for the Y-axis. For the potentially unlimited length of the X-axis, we decided to align the list of monitored databases to it. However, since the iPhone is used in portrait mode by default, the perspective wall has been turned 90°. Now the distorted areas are located at the top and at the bottom of the display, while the focus area maintained its position in the center.

Unlike existing applications using the perspective wall (such as TimeWall [14]), our implementation of the perspective wall is not limited to zoom within the graphical borders of the wall. Thus, the zoom operation may enlarge parts of the displayed data beyond the visual borders of the display. This results in a more appropriate visualization of a so-called drill-down into the information space. Figure 1 shows the four most important visual steps when drilling down on a problem. Each state icon (green = everything is fine, yellow = a warning occurred, red = a critical exception occurred) represents a performance category and each row represents a database. The further a user zooms in, the more information is revealed – first textually, later visually through performance graphs relating to each performance category. At the highest degree of magnification, the user may drill down on a problem, which is indicated by the blue arrows within the performance graphs.

### Interaction techniques

As mentioned previously, we tried to offer the users redundant interaction techniques to allow the selection of a preferred method. Most importantly, we added two new gestures. First, shaking the phone results in resetting the Health Summary back to its original zoom level (in case the user lost orientation within the large dataset). Second, we designed the tap hold/tilt gesture to allow zoom operations single-handed. By default, the iPhone zoom operation is executed through the pinch gesture, which makes the use of both hands necessary. For tap hold/tilt (Figure 1) the user simply taps and holds a finger on the display. Then, he/she may zoom in by tilting the phone to the right or zoom out by tilting the phone to the left. Other interaction techniques are as shown in Table 1: swiping the finger across the display to scroll in one of four directions. Tapping for item selection and tilting in one of four directions to scroll. It is to note that the double tap gesture is not used, since all navigation tasks have been covered by the gestures defined beforehand. Furthermore, no special drill-down gesture has been implemented (e.g. drawing a circle around an item or drawing the letter "L", such as in [15]) since it is hardly

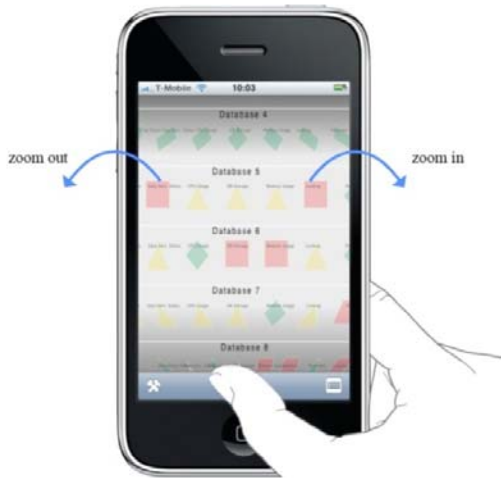


**Figure 1.** The tap hold/tilt interaction technique.

possible to ensure a precise selection of a state icon at the lowest zoom level and the one-handed usability of MultiStates would degrade. This approach contradicts the idea of Nicholson *et al.* [18], where the created application makes use of specifically designed gestures only, thus raising the memory load for each user significantly.

### Configuration through filters

It is clear to see that the display of a mobile device can hardly show all data available. Therefore, and to better support expert users, we introduce several types of filter settings, to reduce screen clutter. Activating these filters results in the hiding and displaying of state types (all states/alerts and exceptions/exceptions only) or performance categories. The setting of such filter options in the preferences panel equals a degree of interest (DOI – as suggested in [16]) function, where the user sets his/her personal area of interest within the application.

Systems, such as LensBar [17] use this kind of functions in order to selectively suppress information. Furthermore the preferences offer the possibility to compress the displayed

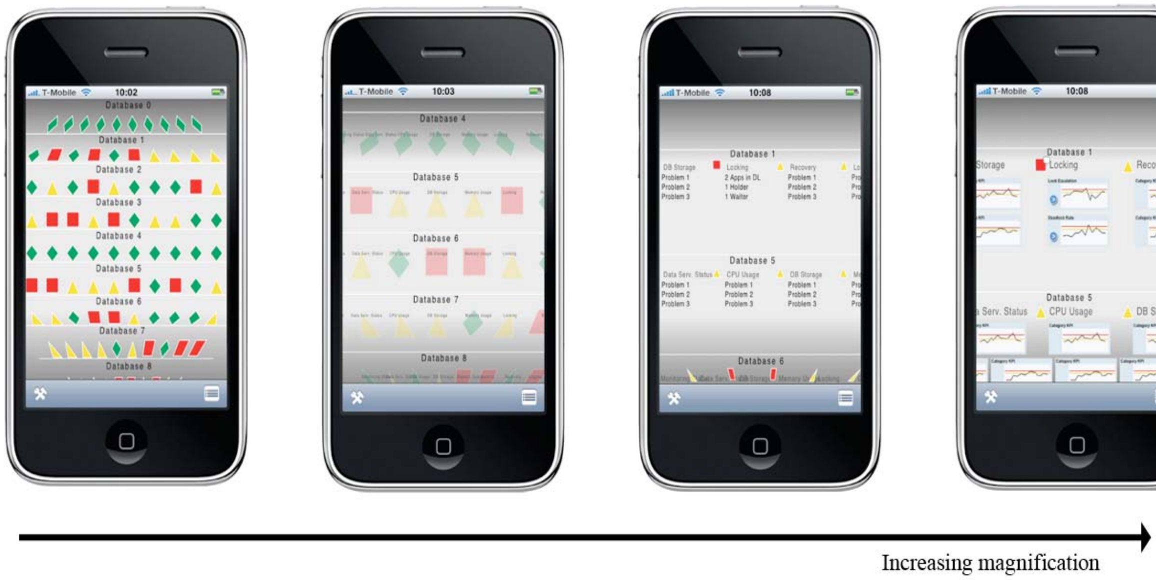


**Figure 2.** The different zoom levels of MultiStates. The magnification increases from the leftmost to the rightmost image.

information by ignoring empty screen space (because of filter settings) and align the database states at the left edge of the display. By filtering for a specific performance category, the user may hide complete databases if no alert or exception has occurred within this category. This may also be accompanied by an audio indicator, which can be used for navigating blindly through the dataset.

| User Interaction | Output |
|---|---|
| Pinch out | Zoom in |
| Pinch in | Zoom out |
| Swipe left | Scroll right |
| Swipe right | Scroll left |
| Swipe up | Scroll down |
| Swipe down | Scroll up |
| Tap | Select an item |
| Tilt up | Scroll down |
| Tilt down | Scroll up |
| Tilt left | Scroll left |
| Tilt right | Scroll right |
| Tap hold and tilt left | Zoom out |
| Tap hold and tilt right | Zoom in |
| Shake | Reset the view |

**Table 1.** Interaction techniques available with MultiStates.

### Support for non-visual exploration

By enabling the audio indicator and a filter category, a sound is played once an alert or exception in this category is displayed at the highest zoom level within the non-distorted central region of the perspective wall (e.g. category “Locking” of “Database 1” in Figure 2).

This indicator can be used in conjunction with tilt navigation to explore the information space without needing to watch the device’s display all the time. Hence, the indicator not only lowers the need to pay attention to the running application for sighted users, but also gives an idea of how to support visually impaired and blind users. However, it is obvious that acoustic feedback may be inappropriate in silent places (e.g. libraries) or noisy environments (train stations, for instance), we decided to make use of an audible indicator for testing purposes and for power saving reasons. Using the vibration control of the iPhone for indicating problems within the Health Summary may lead to decreased battery life when the application is used every day. Even though the creation of an interface for blind users was not the focus of this work, the evaluation results provide insight into further ideas on how to better support both user groups (sighted and blind).

## EVALUATION OF MULTISTATES

We conducted a summative evaluation with six participants in total. Although database administrators are the intended target user group of MultiStates, it was not possible to have DBAs evaluate the system. Instead, we decided to compare three inexperienced users with no professional IT- or database management-related background (Comparison group) with three IT professionals (Expert group: IBM DB2 Performance Expert developer, tester and user experience professional). All members of both groups did not have relevant experience in using an iPhone or other touch-based devices and are sighted.

Each participant was interviewed separately, while being watched by one evaluator. The evaluation was based on two questionnaires: First, a sheet containing tasks to perform using the Health Summary and its filter options. Some tasks were timed and users had to justify why they chose a certain interaction technique while the number of errors made during interaction was counted. Second, a questionnaire focusing on each participant’s usage experience by providing scales to rate the satisfaction and acceptance of application parts (such as the perspective wall, interaction techniques or the complexity of MultiStates compared to a desktop product). Besides the questionnaires, the participants were asked to “think aloud” while they worked with the application. All comments given were transcribed and used for further analysis and interpretation of the questionnaire results.

### Evaluation Findings

To offer a better overview of the results of both groups, we first had a look at each group, before we directly analyzed similarities and differences of both groups. Specific results are shown for each group, while more general results are presented when both groups are compared. Numbers presented in brackets refer to the group’s mean value and the best possible value. Standard deviation is not given due to the small number of evaluation participants (except for the number of corrections needed per group and task).

#### *Comparison Group*

Generally speaking, the Comparison Group provided only high-level feedback. This may be due to the general lack of experience with electronic handheld devices. However, this feedback is especially important for e-Government applications, since these are mostly intended to be used be the “average end user”, who is not familiar with technological details.

*Knowledge in the field of the application domain needed*

The inexperienced users were neither able to judge the capabilities of MultiStates, nor to compare the information available to the depth of information of a desktop product.

*Experience in using the provided platform is helpful*

Some participants had problems interpreting the meaning of system icons on the iPhone platform, since they had not used such a system before. However, not only icons were misinterpreted, but also users were unsure of how to interact with the touch-based device. Participants often asked for help on what they are able to do when performing tasks.

*Pinch and shake are the preferred interaction methods*

Although the overall acceptance of the interaction techniques was high (4.33/5.0) and all techniques were described as intuitive, users did not like the tap hold/tilt gesture and the tilt control. They found both to be too imprecise to work with the Health Summary.

### Expert Group

In contrast to the Comparison Group, the experts were able to provide more detailed feedback on usability problems and interaction methods.

*The perspective wall is suitable for displaying database states*

The expert users showed a high acceptance rate (4.00/5.0) for the perspective wall and mentioned that the availability of context information was helpful when zooming in closely.

*Single-handed use is preferable*

A key criterion of the expert user group was the ability to use the application single-handed. As a result, they preferred the utilization of tap hold/tilt interaction in combination with tilt interaction to quickly switch between scrolling and zooming. Unlike the Comparison Group, the experts described swipe and pinch interactions as precise but too slow to be useful. However, all interaction techniques were described as intuitive to use.

*Complexity requirements have been met*

The results for the question whether MultiStates is a good combination of a complex desktop solution and a mobile application are good (4.33/5.0). Furthermore, the users stated that neither more nor less detailed data is required to be usable for them.

### Comparison

Finally, we looked at similarities and differences between both groups. The most important findings are as follows.

*Speed and accuracy tradeoff*

Looking at more general results, it turns out that all users agreed on the diversion of "slow but precise" input (swipe and pinch) and "fast but imprecise" (tilt and tap hold/tilt). Experts tended to prefer the faster input methods, whereas the inexperienced users liked the more precise techniques better.

*Experts are faster and need fewer corrections*

It turned out that the Expert Group users performed timed tasks faster (Figure 3) and needed fewer corrections than participants of the Comparison Group (CG: 31, EG: 23 corrections – Figure 4).

*Training effect*

Although the Comparison Group needed more corrections than the Expert Group, the number of corrections decreased constantly, except for two peaks in tasks four and six, from task three to task seven. Nevertheless the experts show a more constant number of corrections (based on the lower standard deviation).

*Tilt and shake interaction does not distract from the screen*

Even though some users complained about reflections on the display while tilting the phone, the Expert Group and Comparison Group both did not think that tilting and shaking the phone distracted from working with the application. Especially when shaking the phone, users argued that they knew the result of the action (reset) and therefore did not need to see what happened on the display.

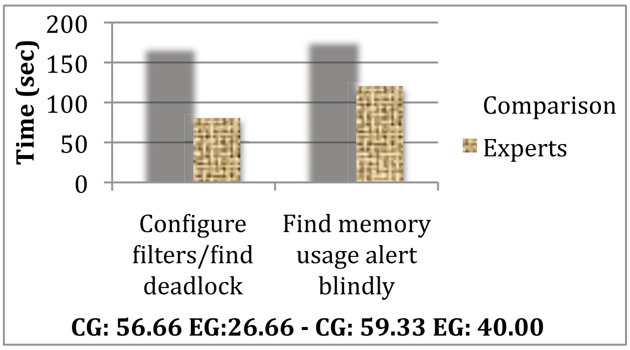


**Figure 3.** The results of the timed tasks (CG: Comparison Group; EG: Expert Group). Average values for both groups are shown below the graph.

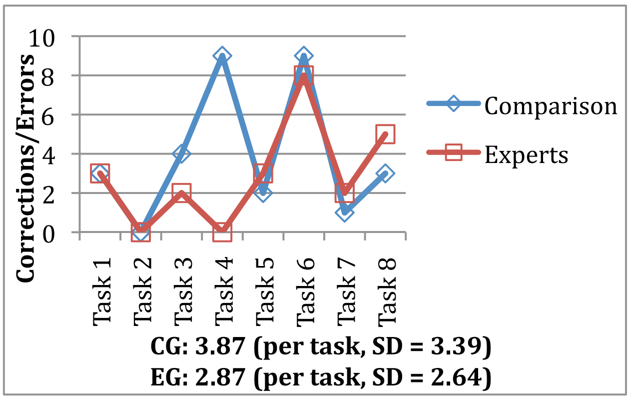


**Figure 4.** The number of corrections needed during interaction. Average values per task are shown below the graph. SD is the standard deviation per task/group.

### Blind navigation feedback

Both groups had to discover an alert or exception in the "Memory Usage" category. To achieve this, they were allowed to configure the preferences menu normally (e.g. activate the audio indicator and set the category filter to "Memory Usage"). As the next step, they had to zoom in and navigate through the databases having their eyes closed. Both Expert and Comparison Group found it neither easy nor difficult to solve this task (CG: 3.33/5.0 EG: 2.66/5.0) although the experts needed significantly less time to complete this task (see Figure 3).

Furthermore, both groups used the tap hold/tilt gesture to quickly zoom in, since it was difficult for them to coordinate two fingers on the phone without being able to see to perform a pinch gesture. Once they managed to zoom in close enough to make use of the audio focus, they just lifted their finger, which was used for tap-hold-zoom, and thus started to scroll through the Health Summary.

## NON-VISUAL EXLORATION: LESSONS LEARNED

All users completed the non-visual exploration and discovery task successfully. However, they suggested improvements for the support of visually impaired people using the perspective wall. Using tap hold/tilt to zoom in and out turned out to be very easy for the users (in this case even for the Comparison Group). But the level of magnification is hidden from the user. As shown in Figure 2, four main levels of information depth are available. Using speech or different sounds as output, these levels may be announced to the user.

Furthermore, the borders of the perspective wall cannot be recognized by blind(-folded) users. They know when they discover a problem they adjusted the filter settings for, but they do not recognize when they reach the edges of the wall when they scroll by tilting the phone.

In addition to the acoustic representation of the current position within the perspective wall, an indicator for the initial position could enhance the feedback. As tilt control may be sensible for slight movements of the phone, it is important at least to know when the phone is held correctly in a way that no action results.

## DISCUSSION

Besides the characteristics of each group described above, it is clear to see that the overall error rate is not optimal, yet. None of the users complained about frustration during the use of MultiStates. Nonetheless, the Expert Group performed significantly better than the Comparison Group in terms of corrections and speed. Independently from the preferred interaction technique of each user, all techniques have been described as intuitive.

Additionally, the acceptance of the perspective wall is high among both groups (CG: 3.33/5.0, EG: 4.0/5.0). Nevertheless, some expert users pointed out that the screen becomes slightly cluttered with all the state information once they zoomed out completely. Creating more visible grids between each database and its states could lessen the cluttering effect.

Errors through accidently tapping the screen have not been counted since the cost of these errors (activating other features or performing different actions) is low. This relates to the definition of mostly tap-independent gestures.

Nevertheless, errors in MultiStates sometimes forced users to repeat an action or to reset the view. The resulting actions have not been added to the number of user corrections.

In the end we were satisfied with the overall acceptance of MultiStates. Especially since none of the users had experience in using touch-based devices before and all of them were able to work with the system after a short demonstration of the interface and the available input techniques

## CONCLUSION

As the introduction described, e-Government is highly dependent on a working IT infrastructure. MultiStates provides support for maintaining this infrastructure. Looking at the main aim of this work, the construction of a mobile and easy-to-use database performance monitoring solution, we can say, that the work was a success. The overall acceptance of it, especially within the Expert Group, was very good and no user failed to complete a certain task.

But the most interesting finding was that even through a simple acoustic indicator, non-visual exploration of an information space can be supported. Even though this is not sufficient to serve as a solution for visually impaired and blind users, it lays out the foundation for further work with touch-based devices.

A platform like the iPhone offers a considerable degree of freedom in creating user interfaces. In particular it offers to integrate the proven scalability of the perspective wall with application-specific set of gestures and allows the display of large datasets on a mobile platform. By offering scroll actions along both X- and Y-axis, we were able to display even more information in combination with geometric and semantic zoom. The flexibility of creating graphical user interface elements independently from hardware buttons and switches is a major advantage and may outweigh missing haptic feedback and lower hit accuracy to some degree.

## FUTURE WORK

Based on the evaluation with IT amateurs and experts we were able to discover leveraging points for the further improvement of the user experience. Most of these points indicate that a longer study with more and different evaluation participants would be helpful in order to improve the usable access of this application.

*Real database administrators for evaluation*

To get more detailed feedback on the MultiStates it would be feasible to have DBAs instead of amateurs and experts evaluate the system.

*Increase the usability for visually impaired and blind users*

By providing more acoustic or tactile feedback, in particular a screen reader or screen magnifier, visually impaired and blind users can be further supported in using MultiStates.

*Refine the display according to accessibility guidelines*

By ensuring a high compliance to accessibility guidelines, such as the Web Content Accessibility Guidelines (WCAG) [22], the user experience of MultiStates may be increased further.

*Prepare MultiStates for iPhone OS 3.0 and higher*

As mentioned above the use of a screen reader (e.g. VoiceOver [23]) and the availability of high contrast graphics are useful features. They are part of the latest iPhone OS, allowing better support for visually impaired and blind users.

*Visually impaired and blind users for evaluation*

Once the enhancements mentioned previously have been incorporated into MultiStates, it would make sense to have blind users evaluate the system and to compare these results to the existing ones.

## ACKNOWLEDGMENTS

We thank IBM Deutschland Research & Development GmbH for supporting this work by making IBM DB2 Performance Expert and IBM Data Studio Administration Console available and for making the expert evaluation sessions possible.

## REFERENCES

1. Salamon, A. Internet Statistics. Available at http://www.dns.net./andras/stats.html#users [Cited: June 23, 2009].
2. Computer Industry Almanac. Computer Industry Almanac Press Release. Available at http://www.c-i-a.com/pr0907.htm [Cited: June 23, 2009].
3. Chaize, I. Sharp increase in the amount of data in the ‘digital universe’: Security Watch – Internet Security News. Available at http://www.securitywatch.co.uk/ 2008/03/13/sharp-increase-in-the-amount-of-data-in-the-digital-universe/ [Cited: June 23, 2009].
4. HKM Projektbüro SAP HR / LUSD. LUSD - Startseite. Available at http://www.lusdportal.hessen.de/ [Cited: June 25, 2009].
5. Deutsches Patent- und Markenamt. DPMA – Startseite. Available at http://www.dpma.de [Cited: June 25, 2009].
6. Date, C. J. Database Usability. *Proceedings of SIGMOD’83.* New York: ACM Press 1983.
7. Jagadish, H.V., Chapman, A., Elkiss, A., Jayapandian, M., Li, Y., Nandi, A. and Yu, C. Making Databases Usable. *Proceedings of SIGMOD’07.* New York: ACM Press, 2007.
8. International Technical Support Organization. IBM Redbooks | DB2 Performance Expert for Multiplatforms V2.2. Available at http://www.redbooks.ibm.com/ abstracts/sg246470.html?Open [Cited: June 25, 2009].
9. IBM Corporation. IBM – IBM Optim Data Studio Administrator for DB2 for Linux, UNIX and Windows - Software. Available at http://www-01.ibm.com/software /data/optim/database-administrator/ [Cited: June 27, 2009]
10. Nielsen, J. Heuristics for User Interface Design. Available at http://www.useit.com/papers/ heuristic/heuristic_list.html [Cited: June 25, 2009]
11. Buring, T.,Gerken, J. and Reiterer, H. User Interaction with Scatterplots on Small Screens – A comparative Evaluation of Geometric-Semantic Zoom and Fisheye Distortion. IEEE *Transactions on Visualization and Computer Graphics,* Vol. 12, pp. 829-836. Los Alamitos: IEEE Computer Society, 2006.
12. Bederson, B.B. et al. DateLens: A Fisheye Calendar Interface for PDAs. *Transactions on Computer-Human Interfaces.* Vol. 11, pp. 90-112. New York: ACM Press, 2004.
13. Buring, T. and Reiterer H. ZuiScat – Querying and Visualizing Information Spaces on Personal Digital Assistants. *Proceedings of MobileHCI’05,* pp. 129-136.New York: ACM Press, 2005.
14. Inxight Federal Systems. Inxight TimeWall. Available at http://inxightfedsys.com/products/sdks/tw/default.asp [Cited: June 23, 2009]
15. Kane, S.K., Bigham, J.P. and Wobbrock, J.O. Slide Rule: Making Mobile Touch Screens Accessible to Blind People Using Multi-Touch Interaction Techniques. *Proceedings of ASSETS’08*, pp. 73-80, New York: ACM Press, 2008.
16. Furnas, G.W. Generalized Fisheye Views. *Proceedings of CHI’86*, pp. 16-23.New York: ACM Press, 1986.
17. Masui, T. LensBar – Visualization for Browsing and Filtering Large Lists of Data. *Proceedings of InfoVis’98*, pp. 113-120. IEEE, 1998.
18. Nicholson, M. and Vickers, P. Pen-based Gestures: An Approach to Reducing Screen Clutter in Mobile Computing. *Proceedings of MobileHCI’04,* pp. 320-324. Heidelberg: Springer-Verlag, 2004.
19. Zhao, S., Dragicevic, P., Chignell, M., Balakrishnan, R. and Baudisch, P. earPod: Eyes-free Menu Selection using Touch Input and Reactive Audio Feedback. *Proceedings of CHI’07,* pp. 1395-1404. New York: ACM Press, 2007.
20. Eslambolchilar, P. and Murray-Smith, R. Tilt-based Automatic Zooming and Scaling Mobile Devices. *Proceedings of MobileHCI’04,* pp. 120-131. Heidelberg: Springer-Verlag, 2004.
21. Mackinlay, J.D., Robertson, G.G. and Card, S.K. The Perspective Wall: Detail and Context Smoothly Integrated. *Proceedings of CHI’91,* pp. 173-179. New York: ACM Press, 1991.
22. World Wide Web Consortium. Web Content Accessibility Guidelines (WCAG) 2.0. Available at http://www.w3.org/TR/WCAG20/ [Cited: June 25, 2009]
23. Apple Inc. Apple – Accessibility – iPhone – Vision. Available at: http://www.apple.com/accessibility/iphone/vision.html [Cited July 23, 2009]