Native Cross-platform Visualization:
A Proof of Concept Based on the Unity3D Game Engine

Markus Wagner,[1] Kerstin Blumenstein,[1] Alexander Rind,[1] Markus Seidl,[1]
Grischa Schmiedl,[1] Tim Lammarsch,[2] Wolfgang Aigner[1]
[1]*St. Poelten University of Applied Sciences, Austria*
[2]*MODUL University Vienna, Austria*
[1]*[first].[last]@fhstp.ac.at,* [2]*[first].[last]@modul.ac.at*

Abstract—**Today many different devices and operating systems can be used for InfoVis systems. On the one hand, web-based visualizations can be used to be compatible with several systems, but the performance depends on optimized browser engines. On the other hand, it is possible to build a native system which supports all the benefits for just one device. However, transferring the code to another system means parts of the code or the programming language have to be adapted. To close this gap, we present a proof of concept based on the Unity3D game engine. We implemented a prototype following the InfoVis reference model and basic interactions for interactive data exploration. A major advantage is that we have now the ability to deploy native code to over 20 different devices. Additionally, this proof of concept opens new possibilities for a future InfoVis framework which benefits from Unity3D.**

Keywords-**Cross-platform; multi-device; native application; Game Engine; Unity3D; information visualization;**

Figure 1.   Interactive data exploration with multiple devices.

I. INTRODUCTION

Visual interfaces, especially Information Visualizations (InfoVis), are high bandwidth gateways for perception of structures, patterns, or connections hidden in the data. Currently, a lot of InfoVis research is focused on domain experts. Yet InfoVis is also increasingly important for a third group, informed citizens and other non-domain-experts.

In recent years, InfoVis has taken essential steps towards the mass market e.g., through infographic websites like Daily Infographic [1], infographics collected on Pinterest by mashable.com [2], or commercialized data exploration tools like Tableau [3] and Spotfire [4]. Pousman et al. [5] introduced the term Casual Information Visualization which complements traditional research in InfoVis with a focus on less task driven activities and a broader audience. This audience also uses a wide range of devices (see Figure 1) whereby the screen resolution differs from 320 x 240 pixels up to 4640 x 1920 pixels [6]. Additionally, Isenberg & Isenberg [7] found "displays ranging from 3.7 inch in the diagonal up to 283 inch". However, the main focus for traditional expert visualization research are desktop computers.

To be prepared for these future trends, cross-platform frameworks become increasingly important. Blumenstein et al. [8] discussed the needs for cross-platform InfoVis frameworks describing three detailed scenarios depending on multiple users, screens and devices. Jokela et al. [9] argued collaborative interaction and connection between devices is a "complex technical procedure". Thomas & Kielman [10] stated four central challenges for Visual Analytics (VA), namely: collaborative analytics, scale independence, information sharing and lightweight software architecture. All these arguments and challenges obviously require a suitable software infrastructure.

Concerning implementation, Baur stated that the choice between web or native code depends on the requirements of the visualization [11]. For big visualization systems, going native cannot be avoided because "in the web it looks like a slide show". Baur also referred to the experience during the implementation of TouchWave [12]. The limitation of native implementations is, that they are only runnable on the device(s) where the code is compiled for. To overcome this restriction, a cross-compiler is needed which supports more than one system. This is where Unity3D [13] comes in, a high performance game engine which supports inter alia 2D and 3D graphics as well as physical simulations. Unity3D code can be deployed to over 20 different operating systems such as desktop, mobile as well as game consoles and

opens the possibility for InfoVis to use all available device features (e.g., camera or microphone access). In contrast to the advantage of cross-platform compilation and high performance rendering, the limitation of game engines is that they do not support any visual data exploration concepts, such as linking & brushing [14], focus & context [15] or small multiples [16].

We contribute to tackle these challenges with a proof of concept based on Unity3D including visual data exploration metaphors to close the gap of suitable software infrastructure for future InfoVis solutions. It supports the creation of interactive InfoVis methods for different devices, different screen resolutions (sizes) and different operating systems, in particular for second screen applications [17].

II. RELATED WORK

A number of software frameworks and toolkits for developing InfoVis applications are available. All of them provide data import/storage solutions and often a variety of widely used visualization techniques. For cross-platform deployment, web frameworks like such as D3.js [18], Protovis [19], or JavaScript InfoVis Toolkit [20] are an evident option and are well extended by various libraries for visualization and data management. Yet, these libraries are often incompatible to each other and impose changes to the coding style that further complicate the already not-so-easy development in JavaScript.

Another cross-platform option are Java frameworks (e.g., Prefuse [21], TimeBench [22], or Improvise [23]). However, Java is not available on some systems (e.g., iPhone OS) or requires specialized SDKs. In addition, none of these frameworks allows for multi-touch interaction or touch gestures which are needed for interacting with mobile devices.

In 2005, Kot et al. [24] started a first approach to integrate information visualization in a game. However, their approach retained the visual metaphors of a first-person shooter showing data items as game-objects in a dungeon-styled environment which is rather different from traditional InfoVis.

Forbes et al. [25] present behaviorism, a framework for dynamic data visualization. Developers can create prototypes fast and afterwards turn this prototype easily into a robust system. However, behaviorism uses JOGL [26] which is outdated. Additionally, Moran et al. [27] presented a two-fold big data visual analytics approach focusing on both, visualization and interaction. As proof of concept they used geo-tagged information of twitter data from MIT's campus and visualized it in the original geospatial domain using Unity3D [13]. Additionally, they provided an interface which responds to natural user interactions to establish a more intuitive environment.

Apart from this, the Tulip 3 Framework by Auber et al. [28] comes with Tulip Graphics, a complete OpenGL rendering engine which was "tailored for abstract data visualiza-tion". The framework is "efficient for research prototyping as well as the development of end-user applications" [28].

In contrast, Sanfilippo et al. [29] described *InfoStar*, "an adaptive visual analytics platform for mobile devices". The conference system provides mobile access to schedules as well as venue information and a "first effective step toward extending visual analytics beyond the desktop paradigm" [29] .

Considering the presented related work, it is apparent that there is a need for frameworks that support cross-platform compilation and touch gestures to be prepared for the challenges and opportunities of the available wide range of interactive devices.

III. PROOF OF CONCEPT

As proof of concept we started with implementing two simple interactive visualization setups to test the Unity3D development environment. Both setups use an open data set of the regional government of Upper Austria on the 50 most often used male and female baby names each year from 2004 to 2013. The dataset includes the variables *name* (nominal), *gender* (categorical), *year* (quantitative) and *count* (quantitative). All these data are merged together into a table and are provided as *.csv file to the visualization. As visualization concepts we combined 1) a circle packing chart [30] with grouped bar charts [31] and 2) a word cloud [32] with a stacked area chart [33].

A. Requirements

To implement such kinds of cross-platform InfoVis techniques based on Unity3D a number of key requirements (R) need to be fulfilled:

- **R1:** Follow the established reference model [34] and other software design patterns [35] for easier adoption in the InfoVis community.
- **R2:** Provide common data exploration and interaction techniques.
- **R3:** Use cross-platform engines to support building for different devices and operating systems [36], [37].

It is important to emphasize that we focus on native and not on web-based solutions in order to have the ability to use all the features provided by the used devices.

B. Circle packing & grouped bar charts

Initially, the circle packing chart shows the first letters of the baby names as bubbles. The diameter of a bubble matches with the number of babies per year. A slider is positioned at the bottom of the screen for selecting the year to display.

By tapping a bubble, the bubble expands and the names which are related to the first letter are shown inside the big bubble (see Figure 2.a). The color of the name bubble is related to the gender (pink := female, blue := male) and the diameter matches the number of babies with this
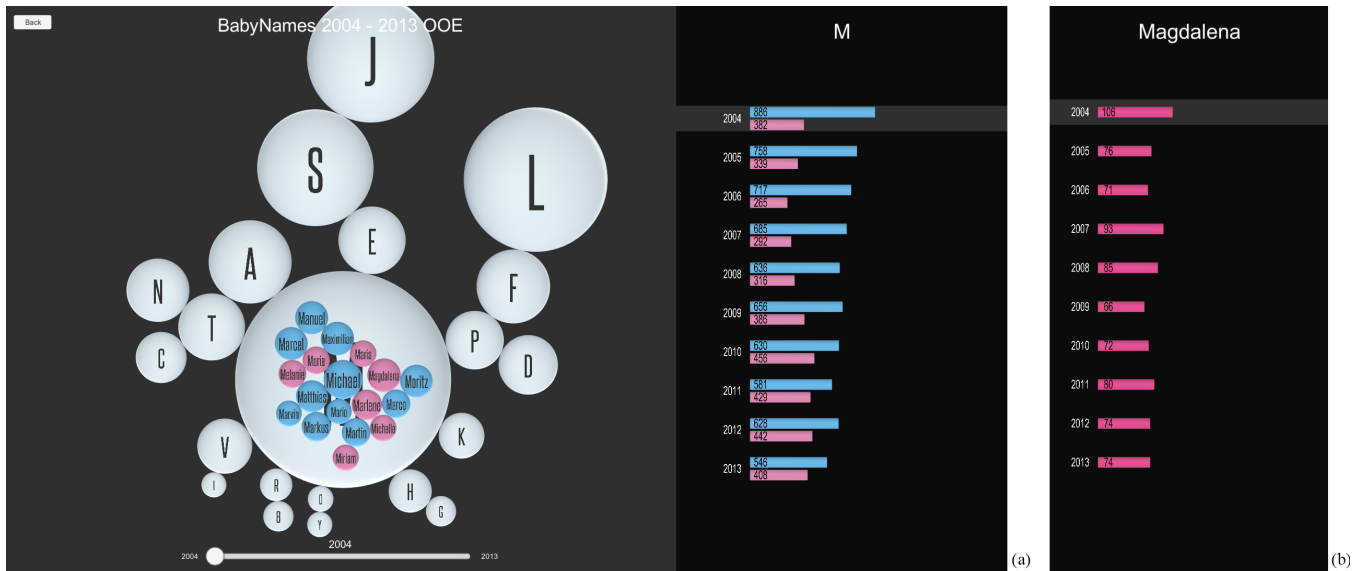
Figure 2. Proof of concept screenshot. a) Shows the circle packing chart with the selected first letter "M" (left view) and the corresponding grouped bar chart (right view) shows the frequency per year for male (blue) and female (pink) names. b) Shows the frequency per year as bar chart for the selection of "Magdalena".

name for the selected year. During the layout phase of the visualization, the bubbles are placed using physics-based movement like gravity and the biggest bubble will be set to the center of the screen. The circle packing chart is linked with a grouped bar chart. The bar chart initially shows the number of babies for all names grouped per year and split into female and male names (using the same colors as for the bubbles). When selecting a first letter bubble, the grouped bar chart shows the number of babies for names starting with the selected letter (see Figure 2.a). When selecting a name bubble, the grouped bar chart changes to a single bar chart presenting the number for the name per year (see Figure 2.b). We also implemented zooming and panning. The user can zoom using a two finger pinch gesture or move the bubbles around using a one finger drag gesture.

C. Word cloud & stacked area chart

In the second visualization, all names are shown as a word cloud (see Figure 3). The font size of a name is proportional to the number of babies over all years and the color matches the gender like in the circle packing chart. The word cloud is linked with a stacked area chart to show developments over time. By tapping on a name, the user can select up to six items in this setting. The highlighted color of the selected name in the word cloud matches the used color in the stacked area chart. For changing the name selection in the word cloud, the analyst has to deselect an item (if the maximum of six elements is reached) before he/she can select another one.

D. Discussion

Based on the implemented proof of concept for our future framework, we fulfilled the presented requirements as follows:

(R1) Follow the established reference model and other software design patterns: For simplifying the handling of the prototypes, we reconstructed the InfoVis reference model [34] in Unity3D. Basically, in Unity3D, each element is a game object which can hold other game objects or scripts, etc. By the use of scripts, it is possible to set various parameters for game object (e.g., rendering, transformations, physics). To replicate the InfoVis reference model, we established a manager object which controls the four stages of the reference model (in the future it should contain changeable game objects for different data and different views). Additionally, we created two separated scenes for our proof of concept. For the interaction through touch gestures, we included canvas elements provided by Unity3D (no separate library is needed). The rendering of the stacked area chart is currently implemented with the low-level graphics library of Unity3D, for all the other visualization techniques, we used included 3D objects (e.g., 3D-text, cylinders, spheres). For the 2D rendering look, we changed the camera settings from 'Perspective' to 'Orthographic'. In general, we only used Unity3D core utilities to realize this proof of concept.

(R2) Provide common data exploration and interaction techniques: Regarding the former reference model based visualization pipeline, we also provide interactive data exploration methods for both visualization settings. Therefore, we implemented linking & brushing [14], zooming, panning, filtering, and details-on-demand [38]. An overview of the
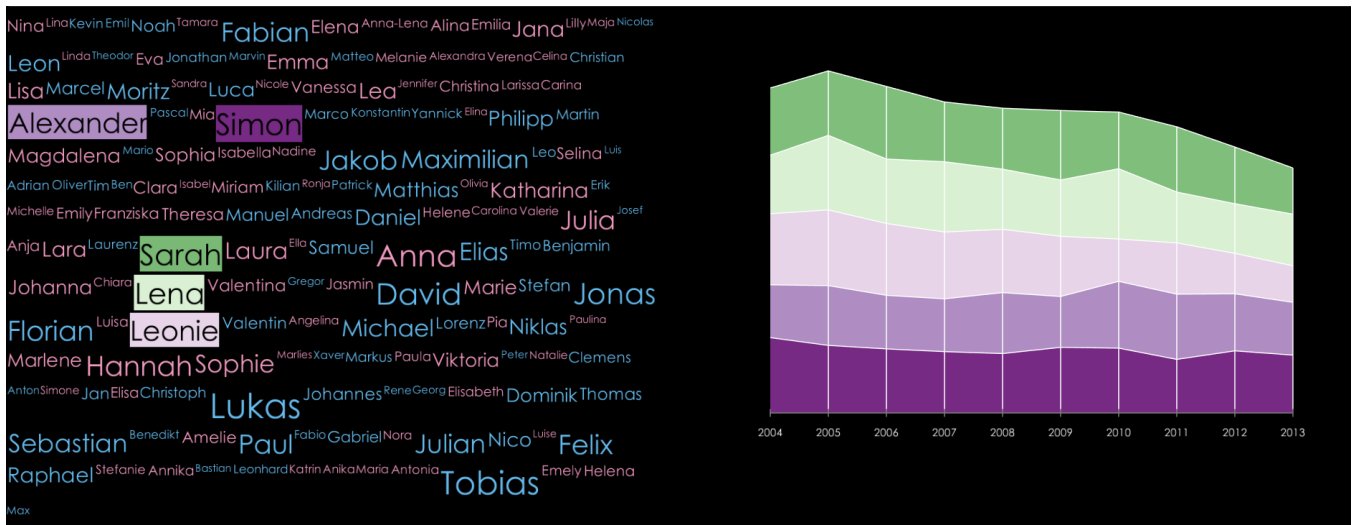
Figure 3. Example word cloud with five selected names and frequency over the years as stacked area chart.

visual mapping and interaction techniques is shown in Table I.

Table I
GIVES AN OVERVIEW OF THE VISUAL MAPPING AND USED
INTERACTIONS TECHNIQUES.

	Circle Packing	Word Cloud	(Grouped) Bar Chart	Stacked Area Chart
Zoom	✓			
Filter	✓	✓	✓	
Pan	✓			
Select	✓	✓	✓	

(R3) Use cross-platform engines to support building for different devices and operating systems: To get the ability for cross-platform compilation, we used the Unity3D game engine which includes a high performance render engine, sounds, physics and more. This engine supports more than 20 different platforms and operating systems including Windows, Mac OS, Linux, Android, iOS, different TV systems, and game consoles [39]. Additionally, WebGL compilation for web page integration is possible. We tested our application on Windows, Mac, Android (tablets) and iOS devices (tablets) and validated its functionality.

By using Unity3D as host engine for our proof of concept, we combined the benefits of a single stable code base and the possibility of native code generation for many platforms and operating systems. This combination allows the use of all available device features (e.g., web cam, gyroscope, touch interaction) which opens new possibilities for future interactive data visualization systems.

IV. CONCLUSION & FUTURE CHALLENGES

In this paper we introduced a proof of concept for a future InfoVis framework based on two concrete InfoVis scenarios. Unity3D provides the ability to deploy to more than 20 operating systems and platforms as native code to use the full performance of the devices and all their included (hardware) features. Depending on the different platforms, we provide appropriate interaction by mouse, keyboard, and touch gestures (e.g., pinch to zoom, dragging).

The creation of such InfoVis frameworks opens up interesting future challenges in relation to interaction and system architecture. Based on the earlier presented key requirements and the work of Blumenstein et al. [8], we defined major challenges to be solved for the creation of such cross-platform environments:

- **Visualization for the masses:** To bring visualization to the general public, it is necessary to develop visualizations that allow to explore the data in an intuitive, easy and understandable way. Users should not have to learn the visualization, instead the should have the ability to gain new insights by the use of the visualization, but also express themselves by showing visualizations casually to peers or audiences [40].
- **Synchronization for collaboration:** Commonly, InfoVis deals with huge amounts of data. A permanent data transfer is very inefficient, because the device resources are limited. Therefore, data structures or query systems are needed for data transfer minimization (e.g., [41], [42], [43], [44]).
- **History for provenance:** Undo and redo functions are a common challenge in the InfoVis community [38], [45], [46]. A novel framework creation is a good starting point to integrate them as core functionalities. Additionally, based on the previously described synchronization approach, it will be possible that every device which is connected to a specified 'network group' has the same history in the backlog. Thus, the analyst gets the ability to change his/her device during performing the analysis

tasks.

- **Interaction and gestures:** A basic set or taxonomy of interactions or gestures should be created to support the experts finding the best fitting technique combination for the used device(s). According to these interactions and gestures there is a second issue depending on the different screen sizes and resolutions of the devices (e.g., lower DPI → bigger gesture; higher DPI → smaller gesture) .
- **Semantic zoom:** Depending on screen sizes and resolutions, the visualization objects can change their shape, details and their presence in the view [47], [48].
- **Support different data sources:** To integrate different data sources and data structures, it will be necessary to develop a generic interface for data input where it is possible to select the data rows or columns of interest for example.

In the future, we are planning to turn this proof of concept into a Unity3D based InfoVis framework, add more visualization techniques, and provide a generic data import structure to generate the possibility to use it as a rapid InfoVis building framework.

REFERENCES

[1] "Daily Infographic," [Online]. Available: http://www.dailyinfographic.com/, Accessed on February 08, 2016.

[2] Mashable, "Infographics on Pinterest," [Online]. Available: https://www.pinterest.com/mashable/infographics/, Accessed on February 08, 2016.

[3] "Business Intelligence and Analytics | Tableau Software," [Online]. Available: http://www.tableau.com/, Accessed on February 19, 2016.

[4] "Data Visualization&Analytics Software – TIBCO Spotfire," [Online]. Available: http://spotfire.tibco.com/, Accessed on February 19, 2016.

[5] Z. Pousman, J. T. Stasko, and M. Mateas, "Casual Information Visualization: Depictions of Data in Everyday Life," *IEEE Transactions on Visualization and Computer Graphics*, vol. 13, no. 6, pp. 1145–1152, 2007.

[6] B. Preim and R. Dachselt, *Interaktive Systeme. Bd. 1: Grundlagen, Graphical User Interfaces, Informationsvisualisierung*, 2nd ed., ser. eXamen.press. Springer, 2010.

[7] P. Isenberg and T. Isenberg, "Visualization on interactive surfaces: A research overview," *I-COM*, vol. 12, no. 3, 2013.

[8] K. Blumenstein, M. Wagner, and W. Aigner, "Cross-Platform InfoVis Frameworks for Multiple Users, Screens and Devices: Requirements and Challenges," in *DEXiS Workshop on Data Exploration for Interactive Surfaces. In conjunction with ACM ITS*, 2015.

[9] T. Jokela, M. K. Chong, A. Lucero, and H. Gellersen, "Connecting devices for collaborative interactions," *Interactions*, vol. 22, no. 4, pp. 39–43, 2015.

[10] J. Thomas and J. Kielman, "Challenges for visual analytics," *Information Visualization*, vol. 8, no. 4, pp. 309–314, 2009.

[11] E. Bertini, M. Stefaner, and D. Baur, "Visualization on Mobile & Touch Devices," 2013, 00:41:49 to 00:46:08. [Online]. Available: http://datastori.es/data-stories-25-mobile-touch-vis/

[12] D. Baur, B. Lee, and S. Carpendale, "TouchWave: kinetic multi-touch manipulation for hierarchical stacked graphs," in *Proceedings 2012 ACM international conference Interactive Tabletops and Surfaces, ITS*. ACM, 2012, pp. 255–264.

[13] "Unity – Game Engine," [Online]. Available: https://unity3d.com/, Accessed on February 08, 2016.

[14] R. A. Becker and W. S. Cleveland, "Brushing scatterplots," *Technometrics*, vol. 29, no. 2, pp. 127–142, May 1987.

[15] J. D. Mackinlay, G. G. Robertson, and S. K. Card, "The perspective wall: Detail and context smoothly integrated," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, 1991, pp. 173–176.

[16] E. R. Tufte, *The Visual Display of Quantitative Information*. Cheshire, CT: Graphics Press, 1983.

[17] K. Blumenstein, M. Wagner, W. Aigner, R. von Suess, H. Prochaska, J. Püringer, M. Zeppelzauer, and M. Sedlmair, "Interactive Data Visualization for Second Screen Applications: State of the Art and Technical Challenges," in *Proceedings of the International Summer School on Visual Computing*. Frauenhoferverlag, 2015, pp. 35–48.

[18] M. Bostock, V. Ogievetsky, and J. Heer, "D3 Data-Driven Documents," *IEEE Transactions on Visualization and Computer Graphics*, vol. 17, no. 12, pp. 2301–2309, Dec. 2011.

[19] M. Bostock and J. Heer, "Protovis: A Graphical Toolkit for Visualization," *IEEE Transactions on Visualization and Computer Graphics*, vol. 15, no. 6, pp. 1121–1128, Nov. 2009.

[20] N. G. Belmonte, "JavaScript InfoVis Toolkit," 2013, [Online]. Available: https://philogb.github.io/jit/, Accessed on May 7, 2016.

[21] J. Heer, S. K. Card, and J. A. Landay, "Prefuse: A Toolkit for Interactive Information Visualization," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, 2005, pp. 421–430.

[22] A. Rind, T. Lammarsch, W. Aigner, B. Alsallakh, and S. Miksch, "TimeBench: A data model and software library for Visual Analytics of time-oriented data," *IEEE Transactions on Visualization and Computer Graphics*, vol. 19, pp. 2247–2256, 2013.

[23] C. Weaver, "Building Highly-Coordinated Visualizations in Improvise," in *IEEE Symposium on Information Visualization*. IEEE, 2004, pp. 159–166.

[24] B. Kot, B. Wuensche, J. Grundy, and J. Hosking, "Information visualisation utilising 3d computer game engines case study: A source code comprehension tool," in *Proceedings 6th ACM SIGCHI CHINZ Conference Computer-Human Interaction*. ACM, 2005, pp. 53–60.

[25] A. G. Forbes, T. Hollerer, and G. Legrady, "behaviorism: a framework for dynamic data visualization," *IEEE Transactions on Visualization and Computer Graphics*, vol. 16, no. 6, pp. 1164–1171, Nov. 2010.

[26] "JOGL - java binding for the OpenGL API," [Online]. Available: http://jogamp.org/jogl/www/, Accessed on May 02, 2016.

[27] A. Moran, V. Gadepally, M. Hubbell, and J. Kepner, "Improving Big Data visual analytics with interactive virtual reality," in *IEEE High Performance Extreme Computer Conference (HPEC)*, 2015, pp. 1–6.

[28] D. Auber, D. Archambault, R. Bourqui, A. Lambert, M. Mathiaut, P. Mary, M. Delest, J. Dubois, and G. Melançon, "The tulip 3 framework: A scalable software library for information visualization applications based on relational data," INRIA, Tech. Rep. RR-7860, 2012.

[29] A. Sanfilippo, R. May, G. Danielson, B. Baddeley, R. Riensche, S. Kirby, S. Collins, S. Thornton, K. Washington, M. Schrager, J. Van Randwyk, B. Borchers, and D. Gatchell, "An Adaptive Visual Analytics Platform for Mobile Devices," in *Proceedings of ACM/IEEE SC*, 2005, p. 74.

[30] J. Heer, M. Bostock, and V. Ogievetsky, "A tour through the visualization zoo," *Communications of the ACM*, vol. 53, no. 6, p. 59, 2010.

[31] W. S. Cleveland and R. McGill, "Graphical Perception: Theory, Experimentation, and Application to the Development of Graphical Methods," *Journal of the American Statistical Association*, vol. 79, no. 387, pp. 531–554, 1984.

[32] F. B. Viegas and M. Wattenberg, "TIMELINES: Tag Clouds and the Case for Vernacular Visualization," *Interactions*, vol. 15, no. 4, pp. 49–52, 2008.

[33] L. Byron and M. Wattenberg, "Stacked Graphs - Geometry & Aesthetics," *IEEE Transactions on Visualization and Computer Graphics*, vol. 14, no. 6, pp. 1245–1252, 2008.

[34] S. K. Card, J. D. Mackinlay, and B. Shneiderman, *Readings in Information Visualisation. Using Vision to Think*. Morgan Kaufman Publ Inc, 1999.

[35] J. Heer and M. Agrawala, "Software design patterns for information visualization," *IEEE Transactions on Visualization and Computer Graphics*, vol. 12, no. 5, pp. 853–860, 2006.

[36] S. Allen, V. Graupera, and L. Lundrigan, *Pro Smartphone cross-platform development: iPhone, Blackberry, Windows Mobile, and Android development and distribution*, ser. Books for professionals by professionals. Apress, 2010.

[37] H. Heitkötter, S. Hanschke, and T. A. Majchrzak, "Evaluating cross-platform development approaches for mobile applications," in *Web Information Systems and Technologies*, ser. Lecture Notes in Business Information Processing. Springer, 2013, no. 140, pp. 120–138.

[38] B. Shneiderman, "The eyes have it: a task by data type taxonomy for information visualizations," in *IEEE Symposium on Visual Languages, Proceedings*, 1996, pp. 336–343.

[39] "Unity – Multiplatform – Publish your game to over 10 platforms," [Online]. Available: https://unity3d.com/unity/multiplatform, Accessed on February 10, 2016.

[40] T. Lammarsch, W. Aigner, S. Miksch, and A. Rind, "Showing important facts to a critical audience by means beyond desktop computing," in *Death of the Desktop—Workshop co-located with IEEE VIS 2014*, Y. Jansen, P. Isenberg, J. Dykes, S. Carpendale, and D. Keefe, Eds., Kasper Anders Søren Hornbæk. Kasper Anders Søren Hornbæk, 2014. [Online]. Available: http://publik.tuwien.ac.at/files/PubDat_233657.pdf

[41] W. Nagel, *Multiscreen Experience Design: Prinzipien, Muster und Faktoren für die Strategieentwicklung und Konzeption digitaler Services für verschiedene Endgeräte*. Schwäbisch Gmünd: digiparden, 2013.

[42] P. Isenberg, N. Elmqvist, J. Scholtz, D. Cernea, K.-L. Ma, and H. Hagen, "Collaborative visualization: Definition, challenges, and research agenda," *Information Visualization*, vol. 10, no. 4, pp. 310–326, Oct. 2011.

[43] T. von Landesberger, A. Kuijper, T. Schreck, J. Kohlhammer, J. van Wijk, J.-D. Fekete, and D. Fellner, "Visual analysis of large graphs: State-of-the-art and future research challenges," *EG Computer Graphics Forum*, vol. 30, no. 6, pp. 1719–1749, 2011.

[44] J. Heer and M. Agrawala, "Design considerations for collaborative visual analytics," *Information Visualization*, vol. 7, no. 1, pp. 49–62, 2008.

[45] C. North, R. Chang, A. Endert, W. Dou, R. May, B. Pike, and G. Fink, "Analytic provenance: process+interaction+insight," in *Proceedings 2011 Annual Conference Extended Abstracts Human Factors in Computing Systems, CHI EA*. ACM, 2011, pp. 33–36.

[46] L. Bavoil, S. P. Callahan, P. J. Crossno, J. Freire, C. E. Scheidegger, C. T. Silva, and H. T. Vo, "VisTrails: enabling interactive multiple-view visualizations," in *Proceedings IEEE Visualization, VIS*, 2005, pp. 135–142.

[47] K. Perlin and D. Fox, "Pad: An alternative approach to the computer interface," in *Proc. SIGGRAPH*. ACM, 1993, pp. 57–64.

[48] M. N. K. Boulos, "The use of interactive graphical maps for browsing medical/health Internet information resources," *International Journal of Health Geographics*, vol. 2, p. 1, Jan. 2003.