Vistribute: Distributing Interactive Visualizations in Dynamic Multi-Device Setups

Tom Horak, Andreas Mathisen, Clemens N. Klokmose, Raimund Dachselt, Niklas Elmqvist
Nowadays, data analysis can take place in many different environments with various devices.
How can we maximize the advantages of multi-device setups while ensuring a minimal user effort?
What we know: devices can fulfill different roles during visual data analysis

Roles emerging from data exploration patterns, e.g., overview+detail, focus+context

Roles emerging from multi-user constellations, e.g., personal toolboxes, shared interaction space

So far:
- Only systems for specific device combinations
- Lacking support for flexibly placing visualizations
- Increasing configuration effort with many devices
What we know: various frameworks for cross-device development exist, but rarely focus on visualizations

**Synchronization frameworks:**
Support for synchronizing elements or events across devices

- Badam and Elmqvist 2014: *PolyChrome*
- Badam et al. 2015: *Munin*
- Houben & Marquardt 2015: *WATCHCONNECT*
- Klokmose et al. 2015: *Webstrates*
- Schreiner et al. 2015: *Connichiwa*

**Distribution frameworks:**
Automatic distribution of components based on manually defined constraints

- Yang & Wigdor 2014: *Panelrama*
- Nebeling & Dey 2016, Nebeling 2017: *XDBrowser*
- Husmann et al. 2018: *Out of Office Software Development*
- Park et al. 2018: *AdaM*

So far, all frameworks...
- rely on additional input from developers or users
- do rarely consider visualization-specific aspects
Visualizations are more than “just” views

Visualizations have a rich body of characteristics and certain relationships to other visualizations.

<table>
<thead>
<tr>
<th>Visualization Type</th>
<th>Encoding</th>
<th>Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data Source</td>
<td>Axis</td>
<td>Internal State</td>
</tr>
<tr>
<td>Data Points</td>
<td>Visual Density</td>
<td>Data Source</td>
</tr>
</tbody>
</table>

Idea: Considering these aspects alongside device properties and user preferences.
We contribute the Vistribute framework

Design Space
Exploring the properties and relationships between visualizations, devices, and the user

6 Heuristics
High-level constraints for deriving a view-sensitive distribution and layout

Vistribute System
Open source implementation representing one possible instance of our heuristics
Each heuristic contributes to different aspects of a distribution

Grouping & alignment based on view relationships

*1 Visual Similarity  *2 Data Similarity  *3 Input Connectivity

View adjustments and device assignments

*4 Data Density  *5 Device Suitability

Allowing adaptations by users

*6 User Preferences
Grouping & alignment based on view relationships

*1 Visual Similarity promotes comparison

* If two views are **visually very similar**, they should be both **juxtaposed and aligned**.

*2 Data Similarity indicates alternative representations

* If two views have a **high degree of data similarity** and a corresponding visual similarity, they should be **placed close to each other**.

*3 Input Connectivity fosters the data exploration

* If an interface component serves as **data input for others**, it should be **placed close to the affected components**.
View adjustments and device assignments

*4* Data Density influences the space requirement

A view should be *allocated space proportional* to the *number of data points* it encodes.

*5* Device Suitability differs for all visualizations

*If devices are diverse, view assignments should be guided by device suitability.*
Allowing adaptations by the user:

6 User preferences always exist

If user preferences are applicable, they outweigh all other heuristics.

Users can have static preference about specific distribution details

In the context of analysis tasks, temporary user interest can occur
Web-based prototype serving as an example implementation
User-created distributions versus Vistribute: a small-scale comparison study

- 6 participants (1 female, 5 male; active in the field > 3 years)
- 2 phases; approx. 60 minutes per session
- Think-aloud protocol

Phase 1:
Manually distributing 10 visualizations in 3 different setups

Phase 2:
Per setup, rating of 3 existing distributions
(2 created by other participants, 1 by Vistribute)
In most cases, multiple reasonable distributions exist

- Personal preferences have a strong influence
- User considered similar aspects as our heuristics
- Manual distributions rated slightly better

<table>
<thead>
<tr>
<th></th>
<th>Manual</th>
<th>Automatic</th>
</tr>
</thead>
<tbody>
<tr>
<td>$M = 3.9$</td>
<td>8.3% 11.1% 19.4% 36.1% 25%</td>
<td>5.6% 11.1% 27.8% 38.9% 16.7%</td>
</tr>
</tbody>
</table>

1 (Unsuitable Distribution)  2 3 4 5 (Optimal Distribution)
Towards effortless multi-device environments

-Manually distributing is “exhausting”, “there should be an optimization for this”

-On average, participants spent 8 minutes on one distribution

Vistribute provides reasonable distributions without requiring additional user input
Towards effortless multi-device environments

- **Next:** Investigating how analysts work in MDEs
  Refinement of heuristics and investigate cross-device interactions

- **From heuristics towards formalism**
  Incorporating AI mechanisms to further improve distributions

- **From distribution towards visualization generation**
  Generating suitable visualization for the user’s current goals
Vistribute: Distributing Interactive Visualizations in Dynamic Multi-Device Setups

Tom Horak - Interactive Media Lab, Technische Universität Dresden – horakt@acm.org
Andreas Mathisen – Department of Computer Science, Aarhus University – am@cs.au.dk
Clemens N. Klokmose – Digital Design & Information Studies, Aarhus University – clemens@cavi.au.dk
Raimund Dachselt – Interactive Media Lab, Technische Universität Dresden – dachselt@acm.org
Niklas Elmqvist – College of Information Studies, University of Maryland – elm@umd.edu

ACM CHI 2019
Glasgow, Scotland, UK