

# Making Sense of Spatial Data: Exploring Visualization Interactions with Architectural Models

Max Kurapov  
University of Calgary  
max.kurapov@ucalgary.ca

## ABSTRACT

This project aims to explore several interactions with spatial data using architectural models. Spatial data can be hard to comprehend, especially when viewed on a computer screen. Scale models placed on digital tabletops can provide tangible interactions to support the exploration and analysis of this data. Using a prototype system, we introduce four interactions: *Reveal*, *Assemble*, *Extract*, and *Extract & Reorient*.

## Author Keywords

Tangible user interfaces, physical models, embodied interaction

## 1. INTRODUCTION

The understanding of spatial datasets has the potential to be improved by anchoring them to physical objects that they represent. If a dataset is describing a large environment, there is difficulty in a coherent understanding of that data, especially when dealing with a large urban area (university campus, downtown). Because this data is usually shown on a computer screen, there can be challenges for viewers with grasping the scale of the environment it is trying to describe, especially given the lack of spatial relationships, and the inability to explore the information from multiple perspectives.

Extensive research has been done to show that physical representations of data can aid in learning and understanding [10].

A better understanding of spatial data can facilitate conversation about the area the data is representing, and most importantly, provide a tangible context in which to perceive the data. An effective merger of tangible objects with a dataset can provide a memorable and immersive experience.

One way to gain understanding of a large environment is by scaling it down, while preserving the relationships between the dimensions of structures. This is done with architectural models. Replica models have a long history, spanning several millennia [3]. When we think of these models in the present, we regard them as tools for prototyping, allowing for tangible interaction, and inviting collaboration. By using models that are easily understood, we can infuse them with their site-specific data and allow for an approachable way of information visualization.

The physicality of the models provides a novel way of interacting with data, and can lead to a better learning experience over traditional methods.

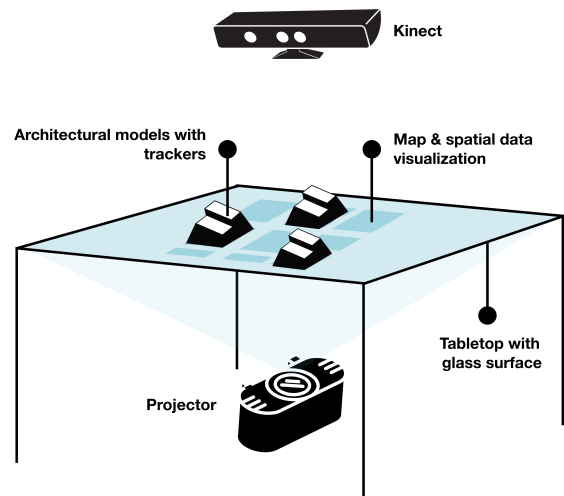


Figure 1. Implementation of the prototype system

There are several ways to manipulate and present spatial data as a result of a user's interaction of physical models. The project will explore some of the interactions that are possible when a system allows for tracking objects placed on a tabletop. We propose four different techniques: *Reveal*, *Assemble*, *Extract*, and *Extract & Reorient*, which users can use to familiarize themselves with a dataset.

To implement these interactions, we created a prototype web-based visualization alongside an existing tabletop interface. The tabletop includes acrylic models of a small urban area: a university campus. In this case, each model corresponds to a building on campus. The buildings on the tabletop have the exact spatial relationships as their real counterparts.

The acrylic models are placed onto a glass surface. It is then possible to illuminate each building separately using a projector underneath the table. This is possible by showing 2D building outlines on the screen via a web-based visualization.

A digital display makes it easy to have different visualizations on the screen. And, by equipping the table top with a tracking device (in this case, a Microsoft Kinect), the models can be manipulated to change the visualization that sits beneath them. As a result, a physical interaction with a small

architectural model can be used to change a visualization of a spatial dataset. The dataset used for this project is of human movement data of students on campus.

The expected contribution of the work is to explore some of the ways users can interact with this tabletop system.

## 2. RELATED WORK

Much of the work done in the area of spatial data representation and physical interface design has been to aid the fields of urban planning, engineering, and architecture. The project aims to be an addition to the research work focusing on using architectural models alongside data visualization [4].

In order to understand the current practices of building these kinds of visualizations, several notable human computer interaction (HCI) research areas should be discussed.

### Tangible User Interfaces

Tangible user interfaces (TUIs) allow for physical objects to be coupled with digital data [6]. The combination of physical models alongside a digital, tabletop interface has been well documented in HCI. Interfaces such as, URP [15], and METADesk [13] couple digital information to physical object. As a result, this allows users to move and manipulate tangible structures that change the underlying data. Similarly to this project, most TUIs make use of tabletops for driving user interaction.

These types of interfaces can increase engagement with the data [9] by providing physical affordances, which are not possible with traditional methods of exploring data visualization (such as a 2D display).

Often, TUIs use physical tokens that do not provide context on their own about their spatial relationships, in contrast to using architectural models.

### Situated Visualization

Situated visualizations display augmented information about the environment that the user is interacting with. One example that makes use of this is SiteLens [16], a system to annotate urban spaces. In this scenario, data is overlaid on top of a real environment using digital screens.

A similar idea is that of embedded data representations: visual and physical representations of data tightly coupled to the real objects and their locations the information refers to [17].

These are general concepts to help formalize the definition of systems alongside different datasets.

### Data Physicalization

Data physicalization is a term used to describe physical artifacts that encode data within their shape or physical properties [8]. While relevant body of work has been done to describe how the practice of architectural model making can aid in the design of data physicalizations [3], this kind of information visualization typically presents abstract data, and not site-specific spatial data.

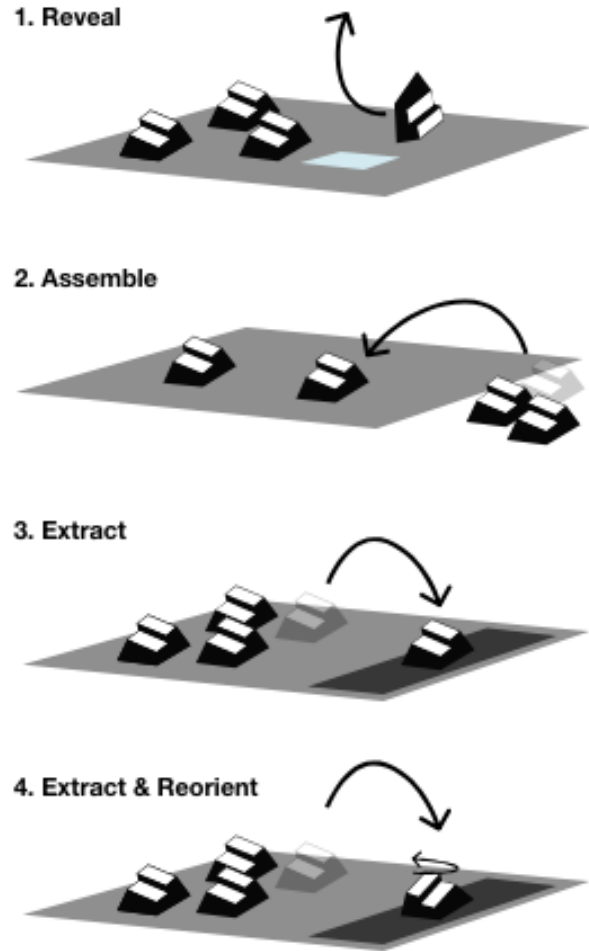


Figure 2. Four interactions

## 3. INTERACTIONS

This project will highlight four possible interactions that exist within the tabletop system, *Reveal*, *Assemble*, *Extract*, and *Extract & Reorient*. Beginning the exploration with four tabletop techniques could be beneficial in developing similar interactions within this design space.

The interactions presented will have several common characteristics, as they exist within the same interface. In general, they can provide benefits over traditional methods of exploring spatial data.

When presented with a tabletop full of small physical models, a user is invited to interact with them given the models' shape and size. Physicality can be a benefit in many tangible user interfaces, as they have the potential to be more accessible and familiar to users (especially novices) given people's knowledge of everyday interactions with objects [12]. Allowing for physical manipulation can make it easier to use such systems, while facilitating learning at the same time [9].

Additionally, the idea of play arises from two factors: through the manipulation of tangible blocks, as well as the use of replica, miniature architectural models, resembling those many people played with as children. The property of playfulness has the potential to increase engagement when learning about novel information.

In terms of interaction design, these ways of manipulating this spatial data align with the idea of fluidity. A fluid interaction in information visualization provides a seamless experience to explore data. These interactions attempt to utilize some of the properties associated with fluidity: promoting flow by providing a user with a sense of control and prompt feedback while using the models, and supporting direct manipulation through quick, reversible, and physical interactions, attached to a single object of interest [2].

Each building's relationship to the real world plays an important role in establishing these interactions as well. By preserving the same characteristics as the actual referent building (like relative size, position, and shape), a better understanding of the associated spatial data can be made, given the model's direct connection to a specific place in the real world.

Finally, these interactions have the potential to encourage and facilitate collaboration, as multiple users are able to explore the dataset through simultaneous actions while moving around a fixed space. 2D displays with a single input device cannot provide the same degree of freedom.

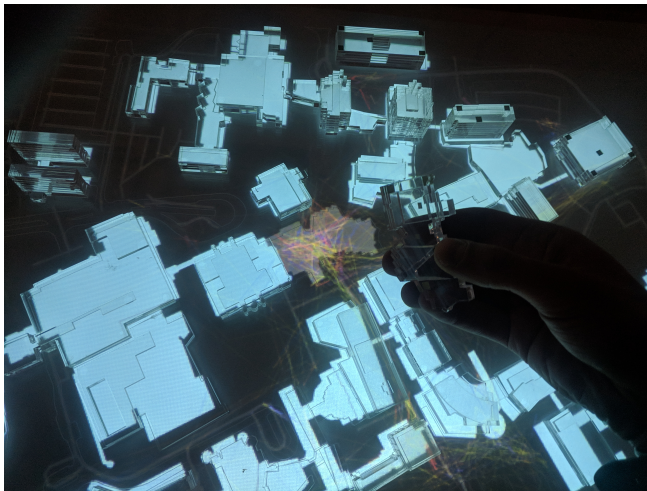


Figure 3. *Reveal* interaction, exposing human movement data for a particular building

## 1. Reveal

*Lift up a building from its place on the map*

This technique begins by placing all of the architectural models in their correct place on the tabletop. The user is presented a top down view of the spatial area that the models and the visualization represent. Having all of the buildings placed on top of the display allows for the user to understand how each piece makes up a part of a whole.

Since each model is small and separate, the user is able to use a simple, embodied motion of lifting up a tangible object off of the tabletop.

As a user lifts up a building, the visualization that is displayed on the screen changes to show the associated data contained within it. Given our data set, this would expose the movement of students through that particular building.

This interaction makes it possible to directly attach a piece of digital data to a physical object, typical of a tangible user interface [6]. A tight coupling of the digital with the physical can be beneficial in the understanding of the underlying data during tangible and tactile exploration [12]. When using traditional approaches to explore spatial data, such as interacting with a 2D map on a screen, this idea of encoding intangible data within a tangible object is lost.

Because the object is an replica model, a user can benefit from knowing what particular building they had removed. This allows for a better sense of the context in which the data is situated in.

As all of the architectural models are in their respective place on the tabletop, each provides the potential of revealing data, and changing the visualization. This kind of interaction can feel like lifting the data out of the digital screen, and a user can have a sense of discovery when interacting with each building. By design, when a user removes a building, only one visual change is reflected in the underlying visualization. This design choice continues on the idea of fluidity, and reduces the likelihood of user confusion through consistent visual feedback.

Additionally, this simple interaction can begin to separate large datasets into more manageable pieces of information. Over time, a richer idea of the dataset can be made as a result of this kind of data exploration [7].

Overall, this scenario allows you to embed a singular piece of data in one tangible object, while providing a strong, contextual experience.

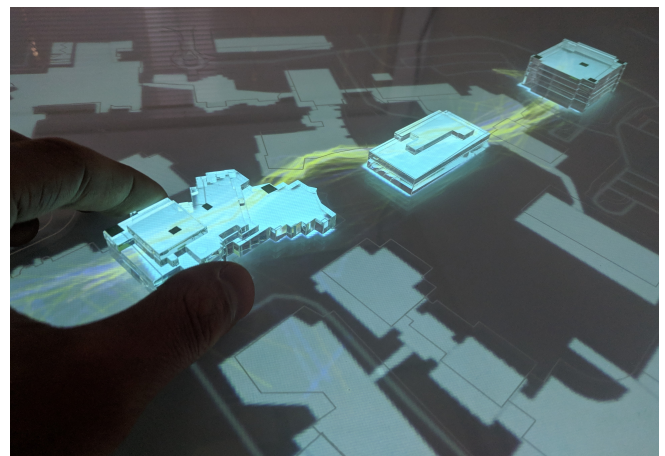


Figure 4. *Assemble* interaction, visualizing human movement between particular buildings

## 2. Assemble

*Grab a removed building, and place it on the map*

This interaction deals with constructing the visualization from a blank state. To begin, the table is cleared of all architectural models, and they are placed on the side. Then, the user is able to place each model corresponding to its digital outline, and as a result, the visualization changes to reflect the new buildings on the tabletop. Given our dataset, this would result in showing human movement data only between the buildings that have been added to the surface.

This interaction was chosen to explore as it provides the user with a sense of constructive control. When all of the buildings are cleared off of the table, the tabletop becomes like a clean slate, a canvas, on which you can begin building on.

Given the many possible configurations of data that can be displayed between buildings, in a sense, the user is able to construct their own visualization. This ties to the idea of constructive visualization.

Inspired by the teachings of Jean Piaget and Seymour Papert, who proposed that children can learn best through manipulation and assembly of objects [11], constructive visualization is a framework that presents how non-experts can create visualizations. [5] The idea of understanding new ideas in this manner can also apply to all people, not just children [1].

Through assembly, and the use of tokens as a basic unit of data within the constraints of an environment (in this scenario, a tabletop), constructive visualization can be used to promote a user's engagement with the dataset through expressive interactions. This idea of assembling architectural models within this tabletop can promote this idea.

Additionally, when the user places the buildings on each of the outlines, they can become more aware of the area that the building takes up within the environment. Mapping the physical shape to its position on the campus has the potential to create a deeper understanding of spatial data.

This way of assembling information could provide a unique way to think about the data in terms of querying. If thinking about the spatial dataset in terms of selecting, this operation is a conditional AND operation. Placing an additional model to the tabletop shows the paths that run through only between the situated buildings.

The idea of using physical tokens to interact with a dataset has been explored by Ullmer et al. [14], but the benefits over a traditional user interface were inconclusive. However, the tokens used in that particular system were abstract, and not architectural models. It is a possibility that the models' relationship to a real place could positively affect a user's understanding of the dataset.

Finally, similarly to the reveal interaction, only one distinct visual change occurs with each single user interaction. This preserves the idea of fluidity, providing the user a quick way to undo their action.



Figure 5. *Extract* interaction, showing building details and associated data in isolation

## 3. Extract

*Remove a building from its position on the map, and place it in a separate, highlighted area*

This interaction makes use of the visualization's side panel visible to the right of the map. A user is able to take a model off of its position, and place it in this highlighted area in order to get information about the particular building.

Beside the model, the building name is displayed. Under it, a circle is drawn. The circle's diameter represents a unit of data, which in this scenario, is the intensity of human movement data at a selected point in time.

This interaction allows for breaking the spatial constraints of the building. Instead of viewing the architectural model as a part of a whole, a user has the ability to highlight the model, and explore its data in more detail. And since the buildings are placed in isolation, the detailed data becomes more coupled with the physical look of an architectural model, potentially leading to better memorability of the dataset.

Multiple models can be placed in this area, allowing to compare data of buildings that are not adjacent in their real world placement. This temporary detachment can support exploration of unique building differences.

By placing the model in separation creates space around the building, and makes possible to place labels around it. When the buildings are in their correct place on the map, labelling is challenging for two reasons: 1) the building's proximity to other buildings, and 2) the refraction of the light through the acrylic material makes it difficult to view the projected text. Providing the side panel is one solution to this problem.

## 4. Extract & Reorient

*Remove a building from its position on the map, place it in a highlighted area, and rotate it.*

Similar to the previous interaction, *Extract & Reorient* makes use of the visualization's side panel, but after placing a building, a user can rotate the model to refine the spatial data that

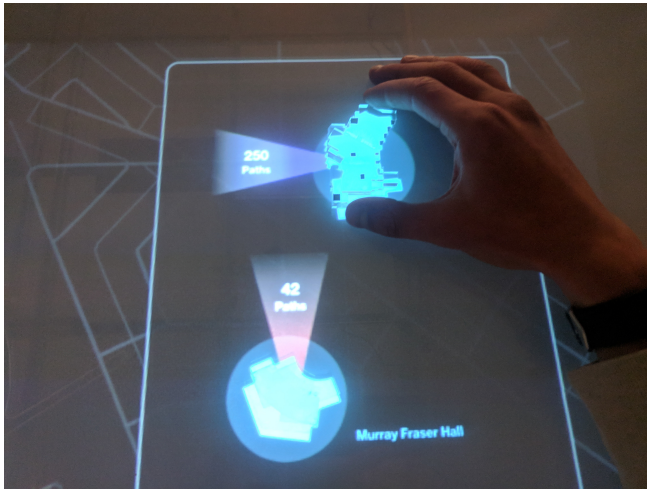


Figure 6. *Extract & Reorient* interaction, filtering data based on building rotation

corresponds to the building. The model acts like a dial in this scenario.

Not only does this interaction allow to break the spatial constraints of the building by placing it in isolation, it also removes the rotational constraint. When a building is on the map, rotating a particular model is difficult to do while keeping the surrounding models in their correct place.

For this implementation, during rotation, an indicator (that is orientating out of the building) displays the model's compass direction. Alongside direction, the indicator protruding out of the model displays the amount of human movement paths that go through the given building. In essence, this interaction provides a way to filter individual building data through another physical action.

Out of all the interactions, this provides the most detailed exploration of the spatial data.

#### 4. IMPLEMENTATION

##### Existing Tabletop and Models

To prototype this system, an existing tabletop was used. The tabletop is made up of a display that is illuminated by a projector from underneath.

Each architectural model is made up of layers of acrylic, joined together using a clear adhesive. The relative dimensions (including height) of each campus building is preserved to provide a better sense of scale of the area represented. In order to track the movement of the models, a Microsoft Kinect is placed above the table top. By placing several squares of reflective material on each model, the models become identifiable to the Kinect. Each model has a unique pattern of these squares. A Unity/C-Sharp application uses the Kinect to distinguish each building from one another by mapping which pattern corresponds to a building. Whenever the position of a building is updated (either by placing it on the table, removing it, or changing its position), the application

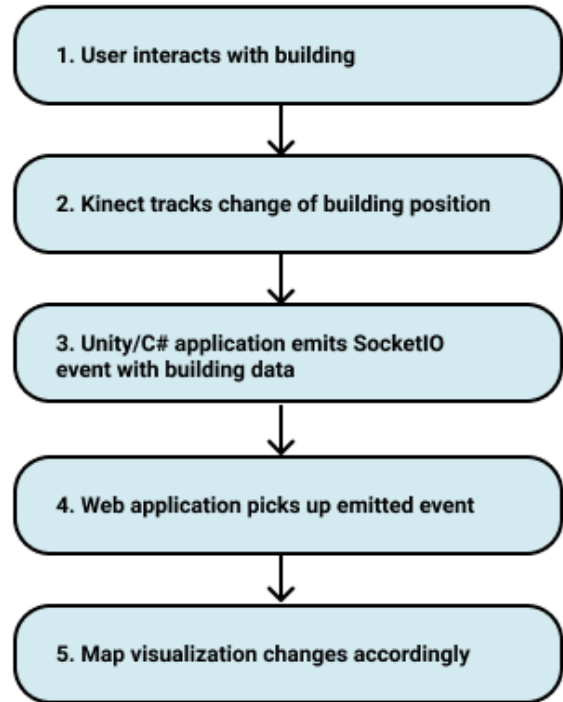


Figure 7. Updating visualization through user interaction

sends out an event with information about the position and the name of the building. This type of real time, event-driven communication is implemented with SocketIO's server-side library.

##### Data

This prototype uses existing, human movement data of students on the University of Calgary campus. 25 thousand location points were extracted from several hundred students who shared their Google location tracking information to the data collectors. This location data from each student was limited only to the university area, and was collected from 2012 to 2017. The data was the anonymized.

By using the timestamp of each point, the points were connected to create individual paths, and as a result, this makes for several thousand student movement lines across campus. Using the latitude and longitude of each piece of data, paths can then be displayed on a map.

##### Browser-based Visualization

For this project, a new visualization was created using web technologies: HTML/CSS and JavaScript. To assist with displaying the building outlines as well as the human movement data, the Mapbox GL JavaScript library was used. After opening a browser window, the visualization is projected onto the tabletop, and illuminates the models from underneath. The visualization shows the student paths overlaid on-top of the 2D building outlines. To the right of the campus map is

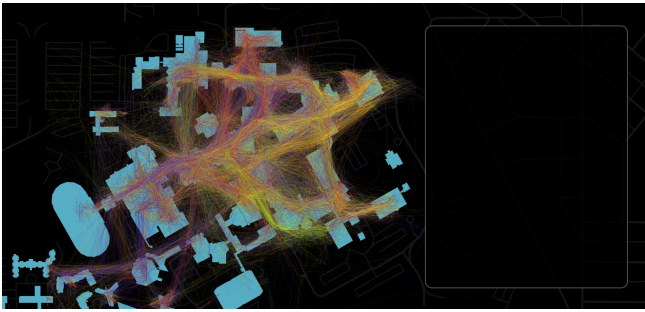


Figure 8. The web-based visualization that is projected under the models, showing all of the human movement paths. The side panel is the dark rectangle on the right.

a rectangular side panel to place individual buildings on, allowing for the *Extract* interactions.

This web interface listens to any events emitted by the Unity application, using SocketIO's JavaScript, client-side library. When a user interacts with a building, the visualization responds to the event, and updates the visible data accordingly.

### Prototyping Interactions

Figure 8 shows the 2D building outlines on the map, that illuminate the acrylic models from underneath. If a building is in its correct spot on the map, the outline is filled in with blue, grey if it is not. On top of the campus map, the human movement data is shown, with different colors representing each path's cardinal direction.

During the *Reveal* and *Assemble* interactions, different human movement paths are displayed depending on which buildings are placed on the tabletop.

For *Extract* and *Extract & Reorient* interactions, the user makes use of the side panel on the right. For the purposes of this project, these side panel interactions were simulated, without complete tracking capabilities.

## 5. DISCUSSION & FUTURE WORK

This system is a proof-of-concept, and not a refined interface. We aim to use this interface in the future to continue exploring this design space, as many different interactions are possible within this area.

After prototyping this system, we see several opportunities for future implementations as well. Most notably, the tracking of the architectural models. While we used a Microsoft Kinect above the tabletop to determine model movement and position (aided with reflective squares on the buildings), other ways of tracking are possible. For example, by placing RFID/NFC trackers on architectural models, or using a different device to track them from the bottom. This should be a consideration when designing similar interfaces.

We believe there are several insights that can be gained from using this system such as: finding out the human movement bottlenecks that exist on campus to aid new planning, helping students compare buildings in which they could study based on the current intensity of human movement, and in general,

help people learn about the campus and how people move through it. All of this data exploration would leverage some of the benefits tangible user interfaces provide, as discussed previously.

Also, the four of the interaction presented should be tested for their effectiveness, with people who have varying levels of familiarity with the campus.

## 6. CONCLUSION

Using a prototype, we have presented four interactions: *Reveal*, *Assemble*, *Extract*, and *Extract & Reorient*, to show how users can use a tabletop interface to explore spatial data using architectural models.

## REFERENCES

1. CHAPMAN, M. *Constructive evolution: Origins and development of Piaget's thought*. Cambridge University Press, 1988.
2. ELMQVIST, N., MOERE, A. V., JETTER, H.-C., CERNEA, D., REITERER, H., AND JANKUN-KELLY, T. Fluid interaction for information visualization. *Information Visualization 10*, 4 (2011), 327–340.
3. HULL, C., AND WILLETT, W. Building with data: Architectural models as inspiration for data physicalization. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems* (2017), ACM, pp. 1217–1264.
4. HULL, C. H., AND WILLETT, W. Data tectonics: A framework for building physical and immersive data representations. *IEEE computer graphics and applications 38*, 5 (2018), 11–17.
5. HURON, S., CARPENDALE, S., THUDT, A., TANG, A., AND MAUERER, M. Constructive visualization. In *Proceedings of the 2014 conference on Designing interactive systems* (2014), ACM, pp. 433–442.
6. ISHII, H., AND ULLMER, B. Tangible bits: towards seamless interfaces between people, bits and atoms. In *Proceedings of the ACM SIGCHI Conference on Human factors in computing systems* (1997), ACM, pp. 234–241.
7. JANSEN, Y., AND DRAGICEVIC, P. An interaction model for visualizations beyond the desktop. *IEEE Transactions on Visualization and Computer Graphics 19*, 12 (2013), 2396–2405.
8. JANSEN, Y., DRAGICEVIC, P., ISENBERG, P., ALEXANDER, J., KARNIK, A., KILDAL, J., SUBRAMANIAN, S., AND HORNBEK, K. Opportunities and challenges for data physicalization. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems* (2015), ACM, pp. 3227–3236.
9. MARSHALL, P. Do tangible interfaces enhance learning? In *Proceedings of the 1st international conference on Tangible and embedded interaction* (2007), ACM, pp. 163–170.

10. O'MALLEY, C., AND FRASER, D. S. Literature review in learning with tangible technologies. 2004. *NESTA Futurelab* (2004).
11. PAPERT, S. *Mindstorms: Children, Computers, and Powerful Ideas*. Basic Books, New York, 1980.
12. SCHNEIDER, B., JERMANN, P., ZUFFEREY, G., AND DILLENBOURG, P. Benefits of a tangible interface for collaborative learning and interaction. *IEEE Transactions on Learning Technologies* 4, 3 (2011), 222–232.
13. ULLMER, B., AND ISHII, H. The metadesk: models and prototypes for tangible user interfaces. In *Proceedings of Symposium on User Interface Software and Technology (UIST?? 97)*, ACM (1997).
14. ULLMER, B., ISHII, H., AND JACOB, R. J. Tangible query interfaces: Physically constrained tokens for manipulating database queries. In *Proc. of INTERACT* (2003), vol. 3, pp. 279–286.
15. UNDERKOFFLER, J., AND ISHII, H. Urp: a luminous-tangible workbench for urban planning and design. In *Proceedings of the SIGCHI conference on Human Factors in Computing Systems* (1999), ACM, pp. 386–393.
16. WHITE, S., AND FEINER, S. Sitelens: situated visualization techniques for urban site visits. In *Proceedings of the SIGCHI conference on human factors in computing systems* (2009), ACM, pp. 1117–1120.
17. WILLET, W., JANSEN, Y., AND DRAGICEVIC, P. Embedded data representations. *IEEE transactions on visualization and computer graphics* 23, 1 (2017), 461–470.