This paper presents data visualizations of an intelligent environment that were designed to serve the needs of two stakeholder groups: visitors wanting to understand how that environment operates, and developers interested in optimizing it. The visualizations presented here were designed for Amatria, a sentient sculpture built by the Living Architecture Systems Group (LASG) at Indiana University Bloomington, IN, USA, in the spring of 2018. They are the result of an extended collaboration between LASG and the Cyberinfrastructure for Network Science Center (CNS) at Indiana University. We introduce Amatria, review related work on the visualization of smart environments and sentient architectures, and explain how the Data Visualization Literacy Framework (DVL-FW) can be used to develop visualizations of intelligent interactive systems (IIS) for these two stakeholder groups.
1. Introduction

Increasingly, our everyday environments are becoming smarter and more connected. IIS is an umbrella term to describe environments that can process data and generate responsive behaviors using sensors, actuators, and microprocessors in order to bridge the gap between humans and machines (1, 2). Sentient architecture implements IIS through artful, imaginative, and engaging architectural artifacts with which we can experiment and observe human behavior and capabilities when confronted with IIS (3, 4). Due to the transformation of the physical environment achieved through scaffolds, sensors, actuators, and microprocessors, new experiences are possible for humans inhabiting these spaces.

Amatria is one such sentient architecture piece on display in Luddy Hall at Indiana University.¹ Foreshadowing a future where the Internet of Things (IoT) is omnipresent, Amatria stands as an artful interpretation of a data-driven environment adapting to what it senses about its physical surroundings. She demonstrates how IoT technologies can affect humans—be it with amazement, unease, mystery, or curiosity. Luddy Hall is a 124,000-square-foot public school building, visited by hundreds of students, faculty, staff, and external guests every day for work, study, talks, and conferences.² Conspicuously placed on the fourth floor of Luddy Hall overlooking the atrium, Amatria is not simply a piece of art seen by hundreds of eyes every day; more


² “Luddy Hall,” Indiana University Bloomington School of Informatics, Computing, and Engineering, https://www.sice.indiana.edu/about/luddy-hall.html
significantly, \textit{Amatria} is embedded into the lives of students, faculty, and staff who strive to develop self-driving cars, explore the social aspects of machine learning, or engineer next-generation robots. In short, \textit{Amatria} embodies a future rich with data where machines and humans live and work together.

The presence of \textit{Amatria} raises the question of how her human cohabitants perceive and understand her. \textit{Amatria} tours have introduced her to elementary children, retirees, invited speakers, advisory council members, wealthy donors, and the president of IU. In order to enrich the experience of each of these stakeholder groups, it is essential to understand their insight needs, meaning what information these groups need to enhance and/or satisfy their interest in the sculpture.

This paper presents research and development efforts on data visualizations that facilitate the communication and exploration of IIS in the form of sentient architecture such as \textit{Amatria}. First, we will introduce related work on the visualization of IIS, and on sentient architecture. Second, we will explain how the Data Visualization Literacy Framework (DVL-FW) can be used to develop visualizations of IIS for two stakeholder groups: visitors and developers. Specifically, we will introduce the Tavola (Italian for “table”) 3D visualization for visitors along with two graph-based visualizations for developers, plus informal user study results that informed the current version of Tavola. Finally, we will discuss lessons learned and planned future work.

2. Related Work

2.1 Visualization for IIS

Previous research on data visualization for IIS is diverse and broad, but a common goal across many lines of research is to provide insights into the observed system, necessitating visualization as an integral part of the system architecture. Common challenges identified in the literature are: system infrastructure, data acquisition, and end-user usability.

Rohaler et al.\textsuperscript{3} implemented a “Cognitive Office” using the Robot Operating System (ROS)\textsuperscript{4} as middleware between a physical office space and a virtual model of it, allowing users to view temperature, humidity, and other data in a 3D model in real time. On the other hand, Jeong et al.\textsuperscript{5} opted for a web-based IoT framework and WebGL\textsuperscript{6} to allow users to control a virtual home.

\begin{thebibliography}{9}
\bibitem{4} ROS, http://www.ros.org
\end{thebibliography}
While the former developed their setup alongside a physical space, the latter simulated input and output of their system, i.e., sensation and actuation, in code only. Furthermore, they presented a two-way communication solution where an end user can both monitor the system and send commands to it. In a user study with 24 participants (14 of whom had some experience in programming), they compared two modes of authoring actions: text-based and visual. They found that user satisfaction, user understanding, user efficiency, and user preference are significantly greater for the visual programming mode. In the area of health management, Hassanalieragh et al. surveyed health monitoring frameworks, particularly noting the need for comprehensive, integrated solutions as a result of the increasing availability of wearable sensors and their data. Similarly to Jeong et al., they emphasize that visualization is needed for end-users (physicians, in this case) to gain actionable insights from heart rate and blood pressure data. Belimpasakis et al. propose a framework for home entertainment using augmented reality, employing cell phone cameras as an interface through which a user can have a two-way communication with an IIS. SensMap features an outdoor view, an indoor view, and a topological view of sensors in a building. This allows the user to view sensor location and data, and link quality to other sensors on a Google Maps basemap in different levels of detail to facilitate data interpretation. As opposed to Roalter et al. and Jeong et al., SensMap uses 2D visualizations only.

Existing visualizations of IIS data help a broad range of stakeholders understand data streams. What’s needed now, however, is a method for systematically designing such visualizations. In this paper, we aim to provide guidelines for designing effective data visualizations that empower different stakeholder groups to explore and optimize IIS.

2.2 Sentient Architecture

Sentient architecture is a type of interactive art. Various approaches from different disciplines exist, including architecture, robotics, and data visualization. In their *Hylozoic Ground* series, Beesley et al. investigate if and how architecture can be perceived as alive by visitors. Building on this, and interested in creating such life-like behaviors, Chan et al. detail the Curiosity-Based Learning Algorithm (CBLA) that implements Intelligent Adaptive Curiosity, as described by Oudeyer et al. As the purpose of CBLA is to increase a system’s behavioral adaptivity through self-experimentation, the

---


8 Jeong et al, “AVIoT: Web-based interactive authoring and visualization of indoor internet of things.”


11 Roalter et al. “A Middleware for Intelligent Environments and the Internet of Things.”


team presented a prototype interactive sculpture featuring an implementation of CBLA using Teensy microcontrollers with ambient light sensors, IR sensors, and accelerometers alongside LED lights and kinetic actuators. Subsequently, Chan et al.\(^ {16} \) tested interactions between human subjects and the testbed. In a user study with 10 participants, they created two test conditions: one with pre-scripted behaviors for the testbed, the other one with CBLA running, switching between modes halfway through the experiment. Performing a quantitative analysis, they found no significant difference in self-reported interest score by the participants between the conditions and noticed only a weak correlation between average activation value and participant interest level. Elsewhere, Börner et al.\(^ {17} \) presented *Lifting the Veil*, a 3D visualization of the *Sentient Veil* sculpture in the Isabella Stewart Gardner Museum in Boston, MA, USA. The purpose of the work was to apply data visualization to sentient architecture in order to satisfy visitor insight needs and help them identify the structure, dynamics, and state of a sentient architecture piece. This work also informed the research and development in this paper.

We consider sentient architecture an instantiation of IIS where two stakeholders exist: visitors who consume (and hopefully enjoy) the art piece, and developers (usually engineers or architects) who build and maintain it. In the following chapter, we will present the various data visualization types implemented for *Amatria*, and we will detail how theory from the DVL-FW was applied during every step of development for each stakeholder group.

3. Sentient Architecture Visualizations

The development of Tavola, a 3D tool for visualizing *Amatria*’s structure and data flow (see *Images 4 - 5*), was guided by a visualization framework, which we will detail in the following section. But the story of Tavola’s creation is also the story of close interdisciplinary collaboration between architects, engineers, software developers, designers, and visualization experts, highlighting the diverse skill requirements needed for data visualization in IIS and in general.

Visualizations usually support two different paradigms of functionality: communication and exploration. Communication refers to instances where visualizations convey insights from data upon which a user can act in an informed manner (such as polls on the news or graphs in a textbook). Exploration, on the other hand, refers to an iterative process where insights from one
visualization lead to questions in need of more insights. Exploratory data analysis, for example, is the process of descending into a dataset by means of multiple visualizations to uncover new questions. Communication is certainly a function of Tavola, particularly in its ability to reveal the hidden structure of *Amatria* to the untrained eye. But even more significant is Tavola’s ability to encourage *Amatria*’s visitors to explore the sculpture more deeply. Tavola lifts the seemingly impenetrable veil of technology for untrained visitors, drawing them into the experience of interacting with a sentient sculpture made out of acrylic, sensors, and microprocessors, forming a coherent and highly technical art piece. In summary, the main goal of Tavola is to encourage visitors to leave their comfort zone and peek ever so slightly behind the curtain of high-tech.

3.1 DVL-FW – Typology and Workflow Process

Our development of sentient architecture visualizations followed the Data Visualization Literacy Framework (DVL-FW) introduced in the *Atlas of Knowledge*\(^\text{18}\) and detailed in a forthcoming paper. The DVL-FW is based on an extensive review of literature from more than 600 publications documenting 50-plus years of work by statisticians, cartographers, cognitive scientists, visualization experts, and others. The DVL-FW process model is shown in Image 2.

The visualization process starts with investigating the insight needs (sometimes also called “task types”) of one or multiple stakeholders. Insight needs dictate which data is to be acquired, what analyses need to be made, and what visualization types are to be chosen. Understanding one’s stakeholders is essential to creating a visualization with actionable insights. Operationalization is needed to guide both data acquisition and the requirement for data cleaning and refinement that comes with it. Analysis then allows the developer of a visualization to reveal patterns and trends in the data, and to refine insight needs and associated questions to the data. Only after these steps are completed can the actual visualization process start. The developer chooses one or multiple visualization types, such as table, chart, graph, map, or network.

Deployment refers to the act of exporting a visualization to a particular medium, such as a piece of paper, a touch screen, a 33-million-pixel video wall, or a virtual-reality headset. Deciding which method of deployment to use is often done early on in the process as some media offer affordances.

more suitable to satisfy user insight needs. For example, visualizations deployed to paper are usually static, fit within a standard book page, and feature annotation in close proximity. Touch screens or keyboard-mouse interfaces, on the other hand, allow users to manipulate their view of the data, or even the data itself. Interactions with data visualizations are only possible if an input device is available and the visualization can receive input in the first place. Interpretation, finally, requires stakeholders to assess what insights can be gained from the visualization with regards to the originally identified insight needs. Very often, results of this interpretation are communicated with a write-up or in annotations to the visualization. After the interpretation is completed, the workflow begins again.

Due to its iterative nature, the DVL-FW process model captures repetition and refinement of insight needs and data visualization not only for the whole series of workflow process steps but also for cycles between just a subset. For example, developers might acquire data based on a perceived insight need, analyze it, and realize that the dataset is not a good fit to provide answers to the stakeholders’ questions—in which case they will likely ask for different data. Similarly, developers might deploy a dynamic visualization and realize that it could be enhanced by adding a second, static visualization. They would then revert to the “Analyze” step to prepare a new chart, graph, map, etc.
3.2 Stakeholders and Insight Needs

While the different stakeholders are quite diverse, we can identify two main groups for which we discuss visualization work here: visitors and developers. Visitors are individuals either unfamiliar with sentient architecture in general, or Amatria in particular, who come to the sculpture either on purpose (such as for a tour) or by chance. Developers are engineers or architects who are interested in how to debug or improve a sentient architecture sculpture.

As outlined in the DVL-FW process model, identifying the rather different insight needs for both types is an essential first step in the DVL-FW process model. While our visualization tool Tavola is primarily geared towards visitors (see section 3.4.1), the data collected in the development process can also be used by developers for debugging purposes (see section 3.4.2).

In order to provide value for visitors, Tavola aims to empower them to explore physical and imaginary spaces. This is called geospatial exploration, a superset of various insight needs usually satisfied with maps. Tavola presents geospatial information from Amatria in a 3D map layout. Maps are among the oldest, most established, and most readable types of visualizations, and Tavola enables visitors to explore Amatria as a physical space. Another insight need addressed by Tavola is comparison, which we achieve by adding a real-time element that shows the consequences of a visitor’s interaction on Amatria. In this iteration, Tavola separates these two insight needs and presents them in different scenes. Finally, Tavola can also be used to identify trends over time, albeit in small units.

Developers, as the second stakeholder group, need to make sure each sensor and actuator functions properly by comparing expected and observed values and behavior. To that end, we developed a real-time bar graph visualization (see section 3.4.2, Image 9) for on-site comparison of IR sensor values. However, developers are also interested in monitoring behavior trends over time in order to find bugs, optimize code, and create improvements based on observations. To this end, we used the same real-time data to develop a graph showing IR sensor values over a 24-hour period (see Images 7 – 8). As the preceding discussion illustrates, it is quite common in the process of data visualization to find that different stakeholder groups are best served by different views of the same data.
3.3 Data: *Amatria* Setup, Software, Networking

In the previous subsection, we investigated insight needs for visitors and developers. This subsection discusses the data acquisition required to implement visualizations for both stakeholder groups.

To satisfy visitor insight need geospatial, a highly detailed parametric model of *Amatria* was shared with the Tavola team. While the original *Amatria* 3D model featured hundreds of thousands of individual pieces, the simplified model was reduced to primitive shapes and 3D objects to represent only the scaffold of *Amatria*, significantly decreasing the performance requirements for whatever target device Tavola would run on. Since *Amatria* took several weeks to install, and since the venue was on the same floor as the offices of the Tavola team, hours of close cooperation and informal exchange were possible. The *Amatria* team helped us understand and prepare for visualization both the model and the real-time data stream (see Image 3, right); additionally, close onsite collaboration during the installation process allowed the Tavola team to gain valuable insights into how *Amatria* functions before it was even finished. Tavola uses the *Amatria* model in two capacities: as a whole, embedded in an additional 3D model of Luddy Hall (also developed by the *Amatria* team based on blueprints from the architects of the venue), and as more high-resolution subset of the model with only parts of the sculpture visible. Planes with opaque material were inserted to replace the transparent walls in the Luddy Hall model, and a highly eclectic, ambient deep-blue light was added to set the building into the background and focus the user’s attention on the sculpture (see Images 4 – 5).

In order to satisfy insight need comparison for both visitors and developers, state information about parts of the sculpture needed to be made available for visualization. IR sensor data was chosen to be streamed from *Amatria* to Tavola by means of universal datagram protocol (UDP). The master laptop (see Image 3) sends commands, retrieves sensor information, and transmits data values to a set of specially assigned IP addresses on the *Amatria* virtual local area network (VLAN). A device running Tavola will get one of these IP addresses upon signing into the network, then receive and parse the data stream. Note that in the current implementation of Tavola, no data is shared about other hardware such as the Raspberry Pis\(^\text{19}\) and Teensy node controllers\(^\text{20}\) that are an integral of *Amatria*’s data infrastructure (see Image 3).

---

19 Raspberry Pi

Specifically, Tavola runs a C# script, creating a UDP client that listens to incoming messages. The master laptop sends out values for all 18 IR sensors at three values per message, with a frequency of 2 Hz. This results in 12 messages per second. The values are packed into a string message and then received in Tavola, where the values are parsed at runtime.

The values arrive in bundles of three because each message contains data for all IR sensors on an individual sound sensor scout (SSS), a distinct part of the sculpture that carries multiple sensors and actuators. Thresholds are set for each IR sensor, which, when reached, trigger a neighbor behavior where the IR sensor’s SSS plays a scripted sequence of actuations. Subsequently, the neighboring SSS perform the same action until the actuation wave has reached the outermost SSS (#1 and #6). SSS #1, specifically, is the focus of scene 2 in Tavola (see Image 5). The message format is to be interpreted the following way: “b” is a prefix added by the Python shell to indicate the presence of the message.

![Image 3: Simplified system architecture (left); raw data stream from Amatria master laptop as logged in Python shell (right). The three numbers at the end of each message are IR sensor values.](image-url)

---

*KATY BÖRNER & ANDREAS BUECKLE*
when printing the incoming bytes and can be ignored. “TV” is a callsign for Tavola, distinguishing visualization-related UDP messages from others on the network. Then follows the SSS for which values are reported, identified by its number (1 - 6). Finally, values for the three IR sensors attached to the SSS are shown.

In order to address insight need trends over time, the same real-time UDP data stream was used. However, while Tavola uses sensor data for live updates, visualizations geared towards developers require data aggregation for an extended period of time. While Tavola parses incoming data at runtime and does not store any data (locally or online), a time series for developers required data storage. Section 3.4.2 details our implementation.

3.4 Analysis, Visualization, and Deployment

In the previous subsections, we investigated insight needs and data acquisition, and illuminated challenges faced along the way. In the following subsections, we detail the visualization process by stakeholder group.

3.4.1 For Visitors

In order to enhance visitors’ understanding of Amatria, we decided to build two scenes in Tavola (see Images 4 – 5). This allowed for an easier-to-maintain application to which we can add more scenes in the future while providing a minimal amount of variety. Scene 1 serves as a structural overview; scene 2 prompts users for physical interaction with the sculpture and enables them to see their own behavior visualized on a small scale. This two-step gave users a gentle introduction to the use of visualization for sentient architecture, while allowing the small visualization team to keep the workload manageable.

We used Unity 3D21, a video game engine increasingly used to develop all sorts of interactive 2D and 3D experiences such as games and architectural visualizations.

Scene 1 addresses insight need geospatial by allowing users to control a virtual camera pointing at the Amatria model, using the physical layout of the sculpture as a reference system. A data overlay consisting of graphic symbols encodes the location of sensors and actuators using graphic variables
(see Table 1). Scene 2 addresses *insight need comparison* where a live sensor value data overlay is updated twice a second over an SSS, allowing visitors to compare that particular sensor’s states based on their own input when waving their hand in close proximity. Similarly, *trends over time* can be identified as the visualization visualizes each data point for exactly 5 seconds, allowing visitors to see whether values increase or decrease.

On the SSS model, apart from the three IR sensors the SSS carries, one microphone sensor, six rebel star lights, six vibration motors, and one speaker are also visible, although no data for them is streamed or visualized at this point. At this point, only sensor readings from IR sensor #2 on SSS #1 are visualized in real-time, encoding human proximity to the sensor via various graphic symbols and graphic variables (see Table 2). A sensor value over approximately 400 indicates the presence of a foreign object such as a human hand; a number under 200 shows the lack thereof. We created a data overlay with different encodings: a particle cone and a proximity index (see Image 5, center-left and lower-left respectively). Tavola is used under SSS #1 (see the location marker in Image 4) to ensure that visitors using the application without a docent understand the real-time character of scene 2, where their physical proximity to the marked IR sensor produces state changes in Tavola. While we only visualize data from one sensor, we are planning to scale up and include more sensor data in the future. Visualizing just one sensor was a good step in developing a visual encoding that can
then be applied to other parts of the sculpture. In a significant investment of time and resources, a dark blue user interface (UI) was developed to complete this as a professionally made and user-friendly visualization.

**Graphic Symbols & Graphic Variables**

Readily available common graphical elements of interactive experiences simplified the implementation of Tavola (e.g., highly adaptable particle systems, various types of lights (point, spot, area, directional) can be created with a few mouse clicks). Similarly, materials can be assigned before and at runtime, prompting us to leverage Unity’s great functionality to create highly custom visualizations with templates for simple and advanced 3D graphics.

In scene 1, to encode the structural data of *Amatria’s* physical layout, we use one graphic symbol (volume) and six graphic variables (see Table 1).

Sensors and actuators are encoded by shape (sphere vs. cube). Each type of sensor and actuator is encoded by one color hue. Each element has a unique position in 3D space. The graphic symbols in the spar field were placed manually. Those in the large and small sphere of *Amatria* (only actuators) were laid out using a custom algorithm that assigned every graphical symbol a 3D location within an invisible spheroid, the center point of which was aligned with the middle of the sphere. Since scene 1 does not contain any
real-time data, the visual encoding is static. Additionally, note that we use color intensity to encode qualitative data (whether parts of the data overlay are turned on or off). Usually, color intensity encodes quantitative data.

In scene 2, to encode the real-time sensor data stream, we use one graphic symbol (volume) and six graphic variable types (see Table 2).

Angle, color intensity, and speed are used for quantitative data encoding. The minimum and maximum values for each graphic variable might be improved by user testing. White and red were chosen as the start and end point of the color intensity gradient, both to ensure maximum visibility against the dark background of the scene and to reflect the red color of the IR sensors in scene 1. The use of x, y, and z-coordinates to encode the location of IR sensor #2 in SSS #1 is analog to how these graphic variables are used in scene 1.

<table>
<thead>
<tr>
<th>Graphic symbol types</th>
<th>Volume</th>
</tr>
</thead>
<tbody>
<tr>
<td>Shape*</td>
<td>Sphere: sensor</td>
</tr>
<tr>
<td>Color hue*</td>
<td>#EF5350 (red): IR sensor</td>
</tr>
<tr>
<td>Color intensity*</td>
<td>Opacity: 0%: graphic symbol turned off</td>
</tr>
<tr>
<td>x-position**</td>
<td>Location of sensor or actuator in 3D space</td>
</tr>
<tr>
<td>y-position**</td>
<td></td>
</tr>
<tr>
<td>z-position**</td>
<td></td>
</tr>
</tbody>
</table>

Table 1: Graphic symbol types vs. graphic variable types in scene 1 of Tavola. *qualitative **quantitative

It is, of course, debatable whether the particle system should be considered a volume (seen as a whole) or as a scatter graph (seen as individual graphic symbols of type point). In the latter case, the graphic variables used would be velocity, color intensity, and speed. We concluded that humans likely perceive changes in the direction of the moving particles as a whole rather than as a collection of points. However, a focused user study could help confirm this.

Also in scene 2, in order to guide visitors to interact with this specific IR
sensor on SSS #1, its current value is visualized within a 2D proximity index, using the graphic variable types of line and pictorial symbol (see Image 5, bottom-left corner).

We compute the x-position of the pictorial hand symbol based on the current IR sensor value. There is no name for this type of visualization, but the 2D reference system and the use of position as graphic variable type for lines of different size could qualify this proximity index as a bar graph. The color intensity gradient of the bars corresponds to the color scheme of the particles. Notably, the proximity index is a 2D visualization within a 3D visualization and introduces purposeful redundancy so that a visitor’s interaction with IR #2 on SSS #1 (in the form of the IR sensor value) is encoded multiple times to maximize visitor understanding when using Tavola unsupervised.

In addition to reference systems and data overlays, annotation plays an essential role in allowing the user to extract meaning from a visualization. In the case of Tavola, annotation is provided in the form of explanatory text and symbols (such as information on how to use the camera, acknowledgements, etc.), accessible via the UI.

Interaction Types

To satisfy visitor insight need geospatial, Tavola uses the interaction types overview, zoom, rotation & panning, and filter. Many interaction types support one of two functionalities: either manipulating the data or manipulating the view. Tavola users can do both. They can manipulate the data
by filtering which parts of the data overlay should be displayed (by means of the toggles in the UI). But they can also manipulate the view of the data. For example, overview and zoom can be used to switch between viewing the whole sculpture and displaying just a small portion of it. In scene 1, rotating & panning the camera enables the user to see Amatria from different viewpoints, especially those that may not be reachable (such as overhead or very up-close). We support these interaction types by putting a gaze lock on the camera, causing the camera to always face an invisible vantage point in the middle of the sculpture, between the spheres and the spar field. Also, the camera is at a fixed distance from the vantage point and moves on the surface of an invisible spheroid, facing inward (a similar gaze lock is in place in scene 2). We should also note that rotating & panning is not part of the DVL-FW but would need to be included to capture the full breadth of 3D visualizations for IIS.

Aside from visualization-related interactions, both scenes require different kinds of real-world interaction between visitor and sculpture due to the fact that this is a visualization of a physical object: observation (scene 1), observation and providing sensor input (scene 2). Through the use of the proximity index and explanatory graphics in the UI, we hope that visitors feel prompted to interact with Amatria and see their influence on the sensors visualized. Whether this works as intended, however, needs to be shown by user studies.

<table>
<thead>
<tr>
<th>Graphic symbol types</th>
<th>Graphic variable types</th>
<th>Line</th>
<th>Pictorial (hand)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Color intensity**</td>
<td></td>
<td>#FFFFFF (white): low sensor value</td>
<td>#D50000 (red): high sensor value</td>
</tr>
<tr>
<td>Size**</td>
<td>Width: 31 pixels ***: low sensor value</td>
<td>Width: 6 pixels ***: high sensor value</td>
<td></td>
</tr>
<tr>
<td>Height***</td>
<td>Height: 0 pixels ***: low sensor value</td>
<td>Height: 14 pixels ***: high sensor value</td>
<td></td>
</tr>
<tr>
<td>x-position**</td>
<td>min (relative): low sensor value</td>
<td>Max (relative): high sensor value</td>
<td>Min (relative): low sensor value</td>
</tr>
</tbody>
</table>

Table 3 Graphic symbol types vs. graphic variable types for proximity index in scene 2 of Tavola. *qualitative **quantitative ***pixels measured in original vector graphic.
Deployment

As mentioned in section 3.1, interactions often create requirements for preceding steps in the DVL-FW process model (see Image 2). Because Tavola allows the user to control camera movement, it was designed to run on touch-enabled devices. Initially, we tested Tavola on a 9.7” tablet. By using a Unity companion app, we could develop the touch interface without needing any time for compilation between versions. As of January 2019, deployment is underway for constructing a kiosk with a 32” touch screen to be placed under Amatria for visitor engagement. Users will be prompted to explore the sculpture in a way that organically adds to the experience of walking under it. This large display will run Tavola for visitors without any supervision or guidance from docents.

Informal Usability Study Results

In order to inform our development process, we frequently conducted informal user studies with team members and visitors during Amatria tours. While subjects had different levels of familiarity with Amatria, none understood the sensors and actuators in the sculpture. Broadly speaking, there were two categories of feedback: feedback on design and feedback on functionality.

The most common remarks we received were on functionality. Upon seeing Tavola running on a tablet, many visitors assumed it was possible to control the sculpture; they interpreted the presence of a 3D model as an affordance for a two-way communication. While this is not part of Tavola, we are in the process of developing such an interface, named “Encefalo” (Italian for “brain”), which is, however, still in a very early stage as of January 2019. We received lots of variations of this particular feedback, such as “I want to see more consequences of my actions on the sculpture,” and “Is the stand Tavola what controls Amatria/acting as her brain?” and “I want to interact with the sculpture via the app.” Comments on design were usually a lot more specific. We readily implemented feedback such as “I should be able to turn specific data overlay elements on and off by touching the corresponding entry in the legend,” and “Attach lines to bottom middle of text labels,” and “Remove [the 3D models of the] focus rooms [next to the Amatria model].” Testing the usability of Tavola is an ongoing effort and will most certainly figure in future research and development.
3.4.2 For Developers

When creating Tavola, we did not have developers in mind as potential stakeholders. For the *Amatria* engineering team which implemented the UDP stream, observing behavior in the sculpture was essential, but individual sensor values were of limited importance, and the team used the Python shell while observing behaviors in the sculpture for debugging.

In an early prototype of Tavola, we visualized IR sensors #1 and #2 on SSS #1 (as opposed to just #2 in the current version). Image 6 shows this prototype in a state when no one is present under the sculpture. The particle cone on IR sensor #2 on the left is white, at a small angle, and the particle...
speed is at its minimum value, which is expected behavior. The cone for IR sensor #1 on the right, however, seems to indicate the close proximity of a foreign object.

Working with the developers to investigate this behavior uncovered an issue with the configuration of that sensor. This led to the realization that data visualization tools can also be helpful for developers, allowing them to “look under the hood” of a complex system.

While Tavola was developed for visitors, in this instance, we could satisfy insight need comparison for developers using the same data, which prompted us to consider this stakeholder group in our visualization design. In the following paragraphs, we will detail two visualizations for developers: a trends-over-time visualization (Images 7–8) and a real-time bar graph (Image 9), along with the visual encoding used for both (see Tables 4–5).

Graphic Symbol & Graphic Variable Types

Interested to understand if this was an isolated bug or the same for all 18 IR sensors distributed over the six SSS, we logged data for all IR sensors to answer other questions about Amatria’s optimal configuration. Over the course of 24 hours in May 2018, we stored UDP messages from Amatria carrying IR values to address the insight need of identifying trends over time. Aiming to see if the thresholds for IR sensors need to be set dynamically based on the time of day (which varied by amount of people in the building, levels of ambient light during the day vs. at night, etc.), we created the graphs in Image 7-8. Here, time is plotted on the x-axis where the leftmost point is the beginning of the observation period (May 7th, ~7pm EST) and the rightmost point is the end thereof (May 8th, ~7pm EST). In order to maintain a manageable dataset size, we plotted only 1% of messages from Amatria, resulting in a message frequency of ~7.2 messages per minute. The data was then grouped into individual tables for each SSS and imported into Processing. In order to create the final visualizations, graphs for all 18 IR sensors were created, and grouped into packs of six to get three visualization sheets. Each of these visualizations show six IR sensors with the same label, consisting of “IR” plus a number from 1 to 3.

Two trend-related insights can readily be gained from these graphs: No threshold change for IR sensors seems to be needed based on differences
between day and night. But more importantly, we found that the consistently high sensor value for IR sensor #1 on SSS #1 is not an isolated incident; in fact, we encountered the same problem for all 6 SSS, specifically in their IRs #1. This actionable insight hinted at a challenging problem for the Amatria and Tavola team that is going to be tackled in the future.

To satisfy the developer insight need comparison, we wanted to allow an on-site team to see state change at sensor level during a maintenance visit in October 2018. This is why we created a real-time bar graph visualizing each IR sensor, grouped by the SSS to which they are attached. Image 8 shows this bar graph at a moment with no person underneath the sculpture; normally, all bars would be expected to hover at 0 or slightly above. What becomes clear is that all IR sensors labeled “IR1” report constantly high values.

Tables 4 and 5 below present the data mappings to graphic variables and graphic symbols used for the visualizations in Images 7 - 9.
Interaction Types

Neither the trends-over-time visualization nor the real-time bar graph require interaction to satisfy developer insight needs comparison and trends over time.

Deployment

To keep the use of the trends-over-time visualization and the real-time bar graph simple, they were deployed on the same 9.7” tablet used for Tavola’s initial development, and a laptop, respectively. The tablet can be carried around under the sculpture when testing IR responses, and the renderings of the trends-over-time visualizations can easily be shared via email or shown at talks and workshops. Thus, deployment meets insight needs.

3.5 Interpretation

The data visualizations presented here differ greatly by stakeholder. User studies will help us evaluate if the visualizations help explore and communicate Amatria’s state and functionality.
Visitors are enabled to explore *Amatria* geospatially, compare IR sensor states based on their own input, and see trends over time for a fraction of the data traveling through *Amatria’s* network. The expectation is that visitors who used Tavola will understand *Amatria* measurably better than those who did not. Since no formal user studies have been conducted yet, we are enthused about the possibility of investigating user interpretations of Tavola (see Section 4).

Developers share some of the insight needs of visitors (*comparison* and *trends over time*), but their requirements for data visualization are quite different. Developers need to ensure *Amatria’s* systems work properly and must check whether observed behavior is expected behavior. Plus they already have a firm *geospatial* understanding of the sculpture. While seeing values for all 18 IR sensor simultaneously in real-time (see Image 9) is a convenient way of ensuring proper functionality of sensors, developers must make decisions based on more information than just sensor values. Hence, to enable developers to make interpretations of the whole sculpture, visualizations with more data need to be implemented in the future.
4. Discussion and Outlook

This paper discussed different visualizations of sentient architecture data: Tavola for visitors and simpler graphs for developers. For all visualizations, we detailed the graphic variable and graphic symbol types, as well as interaction and deployment types. In this last section, we discuss planned research and development.

Going forward, we plan to run formal user studies where a control group of visitors would get a basic oral introduction to *Amatria* and would then be asked questions about its structure, such as “How many IR sensors are there in *Amatria*?” or “Which part of the sculpture is not actuated?” An experiment group would also get this introduction but then explore Tavola before answering the very same questions. Comparison of both user groups—quantitatively in terms of the time needed to complete the questionnaire and accuracy of answers, but also qualitatively based on verbal feedback—will help us understand the utility and usability of Tavola and its impact on how visitors understand and interact with Amatria.

In addition, we are interested to visualize more of *Amatria’s* real-time data, more specifically the interactions between various parts of the sculpture that are not sensors or actuators (e.g., Teensies and Raspberry Pis. During a behavior workshop with *Amatria* and Tavola team members in Toronto in December 2018, alternative data transmission protocols were tested. Open sound control (OSC), already used for communication between the
master laptop and a 4D Sound Laptop (see Image 3), has been found to facilitate the formatting of messages. While investigating OSC for Tavola, we discovered that the workload needed for custom parsing in Tavola can be reduced. Additionally, message queuing telemetry transport (MQTT), already in use in many IoT setups, shows promise in terms of the scalability of data streams out of Amatria. Rather than sending data to specific IP addresses, the master laptop would publish data to a broker that can be accessed by another machine, eliminating the need for the master laptop to be provided with the IP addresses of the destination for the data stream. Another reason for testing OSC and MQTT was to check the feasibility of two-way communication between Amatria and Encefalo, a proposed software to control the sculpture from a graphical user interface. This would enable various stakeholders not only to see Amatria data visualized but also to send input back to the sculpture, resulting in a lot of potential interaction between humans and sentient architecture.

To further advance our line of study, we need to also align our research and development to projects in the realm of the Industrial Internet of Things (IIoT), as well as validate findings with applications and user studies across application domains. While sentient architecture is a fascinating artistic view on smart environments, there are many areas of data-driven innovation that could greatly benefit from data visualization.

As we move forward, we will need to address questions of efficacy: for instance, is there a measurable difference in IIS understanding between groups who use visualizations and those who do not? Questions arise, as well, regarding the DVL-FW. As mentioned, the DVL-FW draws on much previous research on graphic symbols and variables, data scale types, and insight needs: how, for example, can we use 2D visualization within 3D applications (e.g., what is the optimal visualization of the IR sensor proximity in Image 5), or what graphic variables and graphic symbols are useful in 3D visualization? Which interactions are uniquely suited for 2D visualization, and which ones for 3D visualizations, and which can be translated between the two? Furthermore, how can we add interactions that only work on very specific deployment modes (e.g., camera rotation & panning) to the DVL-FW without overloading it?
Acknowledgements

The authors would like to thank Rob Gorbet and Philip Beesley for their expert support of the work presented here; Rob Henderson, Lynne Crohn, Mark Napier for Amatria network IT support; Adam Francey for implementing the Amatria data pipeline; Chauncey Frend for Unity support; Leonard Cross for graphic design support; Todd Theriault for proofreading. Victoria Fard and Timothy Boll kindly provided the Amatria and Luddy Hall 3D model. This work was partially funded by CNS at IU and LASG in Toronto, ON, Canada.

Katy Börner is the Victor H. Yngve Distinguished Professor of Engineering and Information Science in the School of Informatics, Computing, and Engineering, Adjunct Professor at the Department of Statistics in the College of Arts and Sciences, Core Faculty of Cognitive Science, Founding Director of the Cyberinfrastructure for Network Science Center at Indiana University, Bloomington, IN. She is a curator of the international Places & Spaces: Mapping Science exhibit that features large-format maps and interactive data visualizations. She holds a MS in Electrical Engineering from the University of Technology in Leipzig, 1991 and a Ph.D. in Computer Science from the University of Kaiserslautern, 1997. She is a member of ACM and IEEE and is an American Association for the Advancement of Science (AAAS) Fellow and a Humboldt Research Fellow.

Andreas Bueckle is a PhD candidate in Information Science in the School of Informatics, Computing, and Engineering at Indiana University. Coming from a background in video journalism and media, Andreas performs research and development of data visualizations in augmented and virtual reality, exploring the possibilities to visualize and allow for immersive interactions with data in 3D worlds. He holds a B.A. in Media Studies from Eberhard Karls University in Tuebingen and an M.A. in Communications in Economic and Social Contexts from Berlin University of the Arts (Germany).

Additional References

