A Process Model for Dashboard Onboarding

Vaishali Dhanoa1,2, Conny Walchshofer2, Andreas Hinterreiter2, Holger Stitz2, Eduard Groeller4 and Marc Streit2

1Pro2Future GmbH, Austria
2Johannes Kepler University Linz, Linz, Austria
3datavisyn GmbH, Linz, Austria
4TU Wien & VRVis Research Center, Austria

Abstract
Dashboards are used ubiquitously to gain and present insights into data by means of interactive visualizations. To bridge the gap between non-expert dashboard users and potentially complex datasets and/or visualizations, a variety of onboarding strategies are employed, including videos, narration, and interactive tutorials. We propose a process model for dashboard onboarding that formalizes and unifies such diverse onboarding strategies. Our model introduces the onboarding loop alongside the dashboard usage loop. Unpacking the onboarding loop reveals how each onboarding strategy combines selected building blocks of the dashboard with an onboarding narrative. Specific means are applied to this narration sequence for onboarding, which results in onboarding artifacts that are presented to the user via an interface. We concretize these concepts by showing how our process model can be used to describe a selection of real-world onboarding examples. Finally, we discuss how our model can serve as an actionable blueprint for developing new onboarding systems.

CCS Concepts
• Human-centered computing → Visualization theory, concepts and paradigms;

1. Introduction

Data dashboards with visualizations are routinely used across diverse domains [SCB*19], such as health care, education and assembly lines. Dashboards have been defined as visual information displays for monitoring conditions [Few21, WSC17], but recent works suggest that they have a much broader scope [SCB*19]. We use the term dashboard to refer to a combination of multiple data visualizations and textual displays that are often interlinked/interactive and typically arranged in a single-page layout. Most dashboards are initially created by a dashboard author and may then be used by a diverse group of users, ranging from the general public [Lau18] to non-expert dashboard users and potentially complex datasets and/or visualizations, a variety of onboarding strategies are employed, including videos, narration, and interactive tutorials. We propose a process model for dashboard onboarding that formalizes and unifies such diverse onboarding strategies. Our model introduces the onboarding loop alongside the dashboard usage loop. Unpacking the onboarding loop reveals how each onboarding strategy combines selected building blocks of the dashboard with an onboarding narrative. Specific means are applied to this narration sequence for onboarding, which results in onboarding artifacts that are presented to the user via an interface. We concretize these concepts by showing how our process model can be used to describe a selection of real-world onboarding examples. Finally, we discuss how our model can serve as an actionable blueprint for developing new onboarding systems.

Based on the literature [CFGT21], our own experience from previous projects related to onboarding [SGP*19, SWG*21], and discussions with collaboration partners from the steel and pharmaceutical industries who create and use dashboards, we know that the onboarding process can vary based on the target groups (characterized by differences in dashboard literacy). Some users may require in-depth onboarding to a specific visualization type with an explanation of the data sources and the model, while others require only a high-level understanding of the visualizations and how they are linked. Furthermore, onboarding techniques vary in terms of when and where they are applied. Thus, a process model for dashboard onboarding must be sufficiently flexible to correctly describe a diverse range of onboarding approaches. At the same time, it must be specific enough to help developers and researchers to better understand the building blocks of each approach.

The main contribution of our work is a process model for onboarding users to dashboards that fulfills these criteria. In our model, we introduce the onboarding loop alongside the dashboard usage loop (Figure 1). Inspired by the model-view-controller (MVC) software design pattern, we discuss the most important building blocks of the onboarding process. Our model gives rise to an onboarding creation pipeline in which a structured dashboard representation is combined with an onboarding narrative and a set of onboarding means (e.g., human narration, screen recording). The resulting onboarding artifacts, such as oral or video explanations, are presented to the user. We illustrate both the generalizability and specificity of our model by...
showing three different onboarding approaches that can be viewed as manifestations of our model. We also demonstrate an example of onboarding performed using our process model. Additionally, based on our model, we conceptualize a hypothetical interactive and adaptive dashboard onboarding system using an AI model.

2. Related Work

The literature on onboarding is broad, covering multiple domains such as gaming [AOL*12,MLS*16] and software [LGF12]. While onboarding strategies from these domains could be adapted for dashboard onboarding, we limit our discussion of related work to visual analytics. Note, that we focus on onboarding as opposed to guidance. The aim of the guidance is to support users in performing specific tasks with visual analytic tools [CGM*17], such as orienting them during the analysis or providing suggestions about how to arrive at certain insights. In contrast, onboarding helps users to understand, interpret, and use the applied visual analytics methods, providing explanations about key features and capabilities [BMBH16,SGP*19].

For onboarding users to a single visualization, Stoiber et al. [SGP*19] applied the five Ws and H framework to characterize the visualization onboarding space. We use a similar approach to characterize the onboarding space for dashboards in Section 3. In relation to the how, where, and when onboarding is provided, we can further differentiate between active and passive learning approaches [TLM16]. Passive approaches for visual analytics onboarding include cheat sheets [WSMRB20], textual explanations [HH,Rib20], and tutorial videos [Rib20]. Sticky notes [KPS03] are textual callouts that are placed directly within dashboards. They may be passive or active, depending on whether they include a call to action. Fully active approaches onboard users while interacting with a visualization, for instance, by enabling two-way communication [EAB13,BE18], or through context-aware annotations [EB12,KPHH12]. Active approaches have been shown to be more effective than passive ones in specific cases [FDL20, KL16].

Regardless of the exact means chosen, visualization onboarding is often based on a narrative. Stoiber et al. [SWG*21], for example, separated their onboarding instructions for single visualizations into reading the chart, interacting with the chart, and using the chart. In the course of conceptual work for our process model, we realized that an onboarding narrative is particularly important for dashboards, as they include multiple visualizations and potentially complex interactions. This relates dashboard onboarding to (visual) storytelling, a common technique for communicating insights into data to users [BWF*18]. Segel and Heer [SH10] differentiated between author-driven and user-driven stories. In the context of onboarding, we found that author-driven stories created before the onboarding process [LRIC15,BDF15] are used mainly in passive approaches. Active onboarding may also include user-driven elements, for example, when a predefined story is adapted according to the users’ questions and needs. This can be handled via various storytelling techniques, such as Martini glass structure [SH10] and SlideShow [WLF*19]. Also, encoding the interaction history in the visualization guides the users to insight-driven exploration [FPDPH17].

Several attempts have been made to automate the storytelling process [HDR*13,CPL20,SSX*21]. In Section 4.1 we introduce a structured representation of dashboards that may form the basis for automated narrative selection. This representation is closely related to the dashboard meta-models introduced by Vázquez-Inglemo et al. [VIGPT19], augmented with a visualization grammar such as Vega-Lite [SMWH17]. While the dashboard meta-model formalizes aspects of the dashboard itself, we noticed a lack of work on formalizing the onboarding process. We sought to fill this gap with our process model for dashboard onboarding, for which we drew inspiration from other process models related to visualization [FWR*17,BMW17,SGP*19], and from the MVC software design pattern [EB11,ADK*12].

3. Characterization of the Dashboard Onboarding Space

To characterize the dashboard onboarding space, we adapt and refine the Ws and H questions proposed by Stoiber et al. [SGP*19]. Their characterization focuses solely on literacy with respect to a single visualization and does not consider the interplay between the multiple different and linked visual components that form the dashboard. For the purpose of dashboard onboarding, we adjust the Ws and H as follows: (i) **WHO** is the target user, (ii) **WHY** is a dashboard onboarding needed, (iii) **WHAT** needs to be onboarded, (iv) **HOW** is dashboard onboarding provided, (v) **WHERE** is dashboard onboarding provided, and (vi) **WHEN** is dashboard onboarding used. In all of these questions, we refer to both the visual and the functional genre of dashboards, as discussed by Sarikaya et al. [SCB*19].

**WHO** needs to be onboarded is referred to as the dashboard user [TBFGC21] in our process model. Stoiber et al. [SGP*19] describe the characteristics of such a user based on their domain, data, visual encoding, interaction and analytical knowledge. While the same user traits are relevant for dashboard onboarding, we do not make them explicit in our process model as we focus on formalizing the overall process of onboarding. We also highlight the role of the onboarding author, who takes over the tasks related to the creation of the onboarding [LRIC15]. The onboarding author decides the what, how, where, and when of the onboarding process based on the overarching question of why a user needs to be onboarded.

**WHY** a dashboard requires onboarding depends on various factors, such as its data, complexity, purpose, and dashboard user characteristics. The **why** is a driving question behind every choice made in the onboarding phase. We consider this question to be asked before deciding on all the other Ws and H. Therefore, we couple it with the following Ws and H questions to highlight its significance.

**WHAT** can be onboarded in a dashboard includes, but is not limited to, visualizations, global filters, and text elements that provide con-
Dashboard onboarding is a multifaceted problem. While several of its aspects have been touched upon in related work, we found a lack of a solid “scaffold” to support, sustain, and consolidate the discussion of dashboard onboarding in the visualization community. We designed our process model to provide such a scaffold. In order to inspire future work on onboarding authoring, implementation, and evaluation over a wide range of different onboarding strategies, our design process for the model was guided by three requirements. We wanted the model to be:

- **general** enough to cover all existing onboarding strategies without having to introduce new concepts for each particular case;
- **specific** enough to allow detailed description of individual strategies, for instance, by mapping particular elements of existing strategies to general concepts; and
- **actionable**, which means that adhering to it should facilitate the design of new onboarding strategies.

While these requirements may seem obvious, it was not clear how to arrive at a model that would meet all of them simultaneously. We started the design process for the model with an analysis of existing dashboard onboarding strategies. By extracting commonalities and differences of various strategies, and based on related work, we identified fundamental steps and building blocks in the creation and application of the onboarding that we wanted to be reflected in the model. We then iteratively created models and checked whether they met the requirements.

We created several “bottom-up” models based on specific strategies, however, the models generalized poorly. We further created “top-down” models from broad principles but found that in certain cases they require too many modifications, or are too abstract to be actionable. In early versions of our model, we struggled with how to reflect various user roles (see discussion in Section 7.2). Sketches for discarded models can be found in the supplementary material [DWH*19].

After many iterations, we finally converged on a process model that met all requirements to our satisfaction. This model introduces the onboarding loop as an equally important counterpart to the dashboard usage loop (see Figure 1). We based the onboarding loop’s internal structures on the MVC software design pattern [ADK*12]. Our choice to model the onboarding loop in a similar way was motivated not only by aesthetic considerations but also by the actionability requirement.

In the following sections, we first look at the usage loop in more detail and explain how it leads to a structured representation of a dashboard with its components (Section 4.1). We then unpack the onboarding loop, discuss its structural similarity to the usage loop, and derive a “recipe” that describes which parts need to be specified when creating an onboarding system/strategy (Section 4.2).

### 4.1. Dashboard Usage Loop

Dashboards can be described as a collection of typically linked visualizations of a dataset that are controlled by a user through a set of interactions. To formalize this description, we make use of the MVC software design pattern [ADK*12], which allows us to break the dashboard down into three conceptual building blocks (see left part of Figure 2): (i) the underlying data model, (ii) the dashboard’s visual components, and (iii) the control elements or interactions. Typically, dashboards do not reveal the underlying...
Figure 2: Process model for onboarding users to dashboards. In the usage loop, the user manipulates the visual dashboard components through interactions. In the onboarding loop, the user likewise manipulates onboarding artifacts and/or the underlying onboarding content through interactions. The onboarding content is informed by a structured representation of the inner workings of the dashboard.

Data model and logic to the end users, but let them manipulate the data model indirectly through interactions. Such an indirect manipulation of the data model frequently results in an update of the visual components presented to the users. Consequently, it is also possible that interactions directly affect the visual components without altering the data model. We explain the roles of the three building blocks in the onboarding loop in more detail, making use of a simplified version of the New Hires dashboard provided by Microsoft Power BI [SHIS21], shown in Figure 3.

The Data Model broadly consists of the raw data, relations between data subsets, and the transformations applied to the raw data [PC05]. The data model outputs transformed subsets of the data that can later be presented by visual components. For the given example dashboard (Figure 3), the data consists of the business unit, region, ethnicity, and time.

The Dashboard Components are all visual components that are presented to the user. In particular, this includes visualizations such as bar and line charts. Most dashboard components are constructed by applying specific encodings to data subsets which are output by the data model. Details about these components can be listed with the aid of (i) the dashboard meta-model presented by Ingelmo et al [VIGPT19] combined with (ii) a visualization grammar, such as Vega-Lite [SMWH17]. The meta-model for dashboards first enlists all the components required to create a dashboard. Combining the information with the Vega-Lite grammar breaks down individual components into low-level concepts such as axes, marks, and channels. In our guiding example, the main visual components are two bar charts, a line chart, a filter, and the key performance indicator (KPI) at the top left (see schematic in Figure 3).

The Dashboard Interactions are all interactive components in the dashboard that can be controlled by a user through input devices (e.g., mouse and/or keyboard). In typical dashboards, interactions are closely tied to visual components to create an immersive dashboard experience [MM21]. This close relationship has led to ambiguities concerning the view and controller parts of the MVC pattern [ADK*12]. Some interactions, such as filtering or selecting, manipulate the data model. Other interactions, such as zooming, panning, or highlighting, may directly influence the visual components without altering the data model. For instance, filtering through checkboxes is one of the main interactions technique used in our guiding example.

The three building blocks of the dashboard usage loop lend themselves well to a structured representation of a dashboard that can be later used in the onboarding process. The first step is to create a list with all components. For each component, we note the data subset required for its creation and list all its characteristics (e.g., mark type and channels). Figure 3 shows such a representation for the bar chart in our guiding example.

In the next step, we list interactions tied to the visual components. For linked and coordinated views, interactions result in edges between components. Thus, we arrive at a component graph that reflects the relationships between the dashboard’s visual components. In our example, the selection of a bar in the respective chart triggers the filtering of the data used in both the line chart and the KPI. The selection also leads to highlighting in the stacked bar chart. In the component graph, we represent this information as directed edges from the node of the bar chart to the nodes of the other components (see simplified example of filtering components in Figure 3).

To make the component graph useful for the onboarding process, it needs to be enriched with explanations. The explanations can be descriptions of the purpose of the dashboard, information about specific tasks addressed by the components [AES05], or exemplary insights that can be gained by using them. Explanations can also include information about the low-level characteristics of a component, for instance, why a certain mark type or color scheme is used. Such explanations can be provided by the dashboard/onboarding...
The onboarding loop makes use of the information contained in the structured representation explained above and, unpacking it describes how this information is processed to finally present it to the user. Similar to the dashboard usage loop, the onboarding loop comprises three building blocks: (i) the onboarding content, (ii) the onboarding artifacts, and (iii) the onboarding interactions.

The onboarding content consists of the structured dashboard representation together with an onboarding narrative. This narrative drives the traversal of the component graph for the purpose of storytelling, and results in a sequence of components and/or interactions, enriched with their explanations. The narrative is specified directly by the onboarding author or determined by an algorithm. Examples of simple traversal strategies are depth-first traversal (give all or most details for each component immediately) and layout-based strategies (explaining dashboard components in the same order as they appear in the dashboard, e.g., from left to right or top to bottom). More insight-focused narratives may lead to more complex traversals and usually have to be hand-crafted by the onboarding author. A simple depth-first traversal for our guiding example is shown in Figure 3.

In the usage loop, carefully selected encodings are applied to data subsets that are output by the data model to construct the visual dashboard components. In a conceptually similar way, carefully selected onboarding means are applied to the sequences derived from the narrative, which result in onboarding artifacts. As outlined in Section 3, onboarding means can range from human narrations to tooltip information and sticky notes. Depending on the means, the artifacts can be visual, auditory, or textual, for instance.

The onboarding artifacts can be either embedded directly within the dashboard or presented to the user elsewhere. The artifacts are presented depends largely on the means chosen. For example, in the case of video onboarding, the onboarding inter-
face is a video player and the means of onboarding may be a screen recording of the dashboard along with human narration.

Furthermore, the presenter can decide to onboard the user: either before or while interacting with the dashboard. In the case of a programmed guided tour embedded within the dashboard, the onboarding can be retrieved on demand during the dashboard usage; see example in Section 5.3.

The onboarding interactions provide the user with functionality to control the onboarding process. Depending on the type of onboarding, the interactions manipulate different aspects. In static onboarding, interactions merely affect the onboarding artifacts. For instance, in a video tutorial, the script has already been written and the screen has been recorded prior to the onboarding. Only limited controls are given to the user, such as pausing and continuing a video or going back to a specific timestamp/sequence (see example in Section 5.1). In an adaptive onboarding, the interactions trigger updates of the underlying onboarding content. This means that a human presenter may come up with a new onboarding narrative based on a question asked by the user (see example in Section 5.4).

In the exceptional case in which interface and dashboard interactions are directly connected, a user input in the onboarding interface may trigger a change within the dashboard. Conversely, an active onboarding system may listen to events in the dashboard, which may result in updated onboarding artifacts or an updated narrative.

Each block and the connections between them in the general onboarding process model shown in Figure 2 depends on the specific use case.

5. Manifestation of the Process Model in Various Onboarding Scenarios

In this section, we show how real-world onboarding scenarios can be described using our process model. We present four different onboarding strategies—a video tutorial, static textual documentation, a programmed guided tour, and an interactive onboarding session with a human narrator—and discuss the role of each part of the process model in each case. In the first three examples, we analyzed existing techniques and mapped them to our process model. In these cases, we also hypothesize that the component graph represents the mental model of the onboarding author. The fourth example (Section 5.4) describes an onboarding based on our process model, which we carried out with a collaborator. Finally, we describe a hypothetical AI-powered adaptive onboarding system based on our process model. The detailed onboarding process for these examples can be found in the supplementary material [DWH21].

5.1. Onboarding Video

Video tutorials allow the onboarding author to prepare a prescribed story that enables any user to play and recall it sequence by sequence. We screened multiple video tutorials (e.g., on YouTube) that introduce interactive dashboards created for visual analytics tools and chose one that required little domain knowledge. Below we describe a video tutorial using a sales dashboard developed in Microsoft Power BI [Dat21] and show the corresponding process model in Figure 5.

Before recording the tutorial, we assume that the onboarding author chose a narrative based on an example usage scenario of the dashboard. To come up with this narrative, the onboarding author assumed the role of the dashboard user and completed several iterations of the dashboard usage loop. This example usage informed a narrative that covered dashboard overview, filter selection, bar chart explanation with drill-down, and KPIs information. In the case of the video tutorial, the onboarding author decided to record the screen and use an oral narration as the onboarding means. The resulting onboarding artifact is the video, which is presented to the user on the video platform YouTube.

As shown in Figure 5, during the onboarding, the user has the ability to navigate through the video using typical controls such as pausing, fast-forwarding, or jumping to specific sections. While these interactions affect the state of the onboarding artifact that is being shown to the user (i.e., the current video frame), there is no way of triggering an update of the onboarding content. In this sense, onboarding using a typical video tutorial is static, which is reflected by the greyed-out edge between the onboarding interactions and the onboarding content in Figure 5.

The users can choose whether they want to watch the video prior to using the dashboard or whether they want to try to recreate the presented interactions in the dashboard, for instance, while viewing the video on a second screen. There is no explicit connection between the onboarding and the usage loops, which is indicated by greyed-out edges between the user and the dashboard usage loop (in Figure 5).

5.2. Textual Explanation

For the second example, we used the regional information dashboard about climate change projections from the IPCC WGI Interactive
The programmed guided tour is presented as a text document (see Figure 6). Based on this description, the traversal of each component, as the sequence with its required components (WHAT) is fixed and programmed in a way that cannot be adapted. Onboarding is represented using the means of highlighting and callouts or tooltips (HOW), which are overlaid on the dashboard itself (WHERE). The programmed guided tour is presented as a simple stepper interface when accessing the dashboard for the first time (WHEN). It can be accessed at any time when interacting with the dashboard by clicking on the information icon.

The stepper interface lets the user navigate the sequence of explanations, allowing them to either go back to the previous step, proceed to the next one, or end the onboarding process. This predefined onboarding narrative and the resulting sequence cannot be customized by the user. It is reflected by a grayed-out edge between the onboarding interactions and the content in the process model (Figure 7). During the onboarding process, the usage of the dashboard is either blocked or limited and has no effect on the onboarding narrative. For the most part, this leads to disconnected usage and onboarding loops. In only one case does the interface explicitly encourage the user to interact with the dashboard by selecting a region the map to continue the onboarding process. This one-time interaction is indicated in the process model (Figure 7) by the edge from user to select region in the dashboard. This selection triggers the onboarding stepper interface once and the onboarding process continues. The subsequent onboarding steps are again controlled through the overlaid stepper interface.

Upon restarting the onboarding process during dashboard usage, the user has to start the predefined onboarding story again from the first component, regardless of the already introduced components. The system does not save any past interactions with the dashboard or the onboarding, and the traversal order of the component graph is fully fixed.

5.4. Interactive Onboarding with Human Presenter

For this example, we authored an onboarding for an interactive dashboard based on our process model. We presented it to a dashboard user in the course of a collaboration with the steel industry.

In the previous examples, onboarding was designed for heterogeneous groups of users. In this case, we present an onboarding that was tailored to specific users who had comprehensive data and domain knowledge. They were not new to interactive dashboards overall, however, a beginner in the field of visualization. For this purpose, we created a list of components that required onboarding (WHAT) from our experience. We then listed two possible predefined sequences for explanation and chose the one which increased the complexity of onboarding gradually. The onboarding took place virtually using screen sharing via an internal communication platform for teleconferences (WHERE). The narrative was provided to the user in the form of oral explanations (HOW). As we made use of visualizations in which the user had little to no prior experience, we provided the onboarding before the first-time use of the dashboard (WHEN). This scenario is reflected in the process model (Figure 8) by the lack of a “use” connection from the user to the dashboard, while the “presented to” connection from the dashboard components to the user still exists.

After our explanation, the user posed questions to us, which is represented by the Interactions block in Figure 8. They also used the shared screen to annotate the visuals for clarifying the explanations provided by us. The user’s questions either (i) prompted a change in the oral explanation, which led us to rephrase what was said without changing the onboarding strategy; (ii) caused us to update the underlying onboarding model, for example, by changing the narrative to explain components of interest; or (iii) triggered an interaction in the dashboard, potentially requiring an update to the narrative to reflect the resulting changes.
After one hour of onboarding, the user was done posing questions on various visualization types and interactions. At this point the user had gained an understanding of how to derive actionable insights from the dashboard, indicating a successful session. We discuss the extent to which the process model helped us in the authoring and evaluation of our onboarding in Section 6.

5.5. AI-based Onboarding

Finally, we applied our process model to a hypothetical onboarding scenario, which tries to overcome the existing flaws of human-narrated onboarding scenarios described in Section 5.4. One of its obvious flaws is that, once the onboarding has been completed, it cannot be repeated or accessed on-demand. Further questions can be asked by sending requests to the onboarding author, but the answers may not be provided immediately. Thus, we imagine an AI-based solution to this problem that can be represented by our process model. In contrast to the human narration, such an AI-powered onboarding can be accessed by the user flexibly at any time.

In our proposed automated onboarding solution, the user can initiate the onboarding process by typing in a query such as “On-board me to the dashboard”. This triggers an AI model to derive a component graph from the dashboard with some example insights. With the help of this component graph and a predefined, initial traversal strategy (e.g., depth-first), the AI model can create an onboarding narrative. This narrative may be presented using callouts or tooltips (HOW), overlaid directly onto the relevant dashboard components (HERE). The user can then choose to follow the predefined narrative, skip steps, or stop it by interacting with the callouts and/or the dashboard. When the user poses a question to the interface, an AI model could respond in ways that resemble a human reaction by (i) rephrasing the current explanation, (ii) changing the narrative, or (iii) initiating changes directly in the dashboard. Possible strategies for updating the narrative could be based on the user’s previous interactions with the dashboard and/or the onboarding interface. For parsing the questions and relating them to individual dashboard components, we envision a natural language processing model similar to LILY [QDRH19], which was developed for creating dashboards from written prompts.

The process model for such an AI-based onboarding looks like the one for an interactive session with a human presenter (Figure 8), except for two additional edges. First, since such an onboarding would take place synchronously with the dashboard usage, the edge from the user to the dashboard is present. Second, as the AI model can update the onboarding based on interactions between the user and the dashboard, the second, left-to-right “trigger” edge between the interactions would be present. As a result, this hypothetical AI-powered onboarding process would make use of all connections in the general model shown in Figure 2.

6. Using the Process Model

In a data-driven world, the use of dashboards is inevitable. As highlighted by Sarikaya et. al [SCB*19] and Tory et. al [TBFGC21], it is important to support the end users in dashboard literacy. Answering their call to action, we provide practical advice to those who want to support users in understanding dashboards. Accordingly, in this section, we discuss ideas and insights gained from onboarding in Section 5.4 by applying our process model for dashboard onboarding in practice. We group these ideas into three phases that reflect the onboarding process: authoring, implementation, and evaluation.

6.1. Onboarding Authoring

The authoring phase of a dashboard requires the author to choose subsets of data to be visualized and presented to the user. Similarly, the authoring phase of the onboarding requires the author to select specific visual components and interactions that should be explained to the user.

Previous approaches aimed to assist the author in the dashboard authoring phase by selecting the needed data subsets and/or visualizations [THHS16]. However, there is insufficient literature on how to author the onboarding phase. In this section, we describe how our contribution can potentially facilitate various aspects of the onboarding authoring.

The structured representation, introduced in Section 4.1, can be made explicit by the onboarding authors to identify dashboard components, interactions, or pieces of data-related information that may be vital to the end user. For the onboarding case in Section 5.4, we used the structured representation to gain the first idea of narration sequences that arose “naturally” from the dashboard structure. It indicated that the time slicer had the most relations to the other components. This helped us in deciding two possible narrative sequences, where the slicer could either be introduced first or towards the end.

Figure 7: Onboarding with a programmed “guided tour”, described using our process model.
We decided to go with the latter to gradually increase the complexity of onboarding. Additionally, attaching small explanations to individual nodes in the component graph allows the author to break down a potentially complex onboarding narrative into manageable parts. This may also enable the author to reuse individual explanations, regardless of the specific narrative chosen or onboarding means applied later in the process.

Finally, the onboarding recipe, shown in Figure 4, describes how a specific set of onboarding artifacts can result from choices made during the authoring phase. Even though the choices of narrative and means may be interdependent and strongly coupled to the overarching question of WHY an onboarding is needed, the recipe reveals that one onboarding strategy may be transformed into another by choosing alternative means. For instance, using the onboarding recipe, it should be straightforward to create a programmed guided tour based on textual documentation.

6.2. Onboarding Implementation

The structure we chose for both the usage and the onboarding loops, is closely based on the MVC software design pattern. Thus, our process model can serve as a guide in the implementation of new onboarding systems. For example, it can be used to infer when the dashboard needs to send information to the onboarding interface or vice versa. More importantly, the process model can help onboarding authors identify when an update of the underlying narrative is necessary and when updating only the onboarding artifacts is sufficient (e.g., choosing a different explanation strategy for the same visual component).

Since the structured representation is the foundation of the onboarding loop, we present our thoughts on how to automatically extract this representation. The first step is to extract the components of the dashboard and their relationships. For an embedded Microsoft Power BI dashboard [BSB+21], Embedded Report APIs, such as Power BI Client API [Mic21], can be used to this end. We assume that similar workflows are viable for other tools such as Tableau [Tab21]. In the second step, the enrichment of the structured representation with explanations must be formalized. This can easily be specified through an approach similar to Encodable [Won20], a configurable grammar for visualization. Further thoughts on how to make use of Encodable can be found in the supplementary material [DWH+21]. After the creation of the fully enriched graph, any graph traversal strategy can be implemented as a default strategy for the onboarding. The resulting sequence can then be shown as a textual description using callouts, overlaid directly on the dashboard for the user. In the case of adaptive onboarding, the onboarding controls can allow the user to skip or access parts of onboarding based on their needs. Note that this is only one way of implementing the process model, and the capabilities of the implemented model can be enhanced in various ways, for instance, by inferring user preferences.

6.3. Onboarding Evaluation

Section 5 lists several examples of real-world onboarding processes described with our process model. We believe that taking existing onboarding sessions and phrasing them in terms of the process model can help presenters of dashboards to identify pain points, either regarding the dashboard usage or the quality of the onboarding. In case of Section 5.4, we recorded the session to analyze when it would have been more appropriate to show something directly in the dashboard or when it would have been more effective to update the narrative based on the user’s needs. This helped us to improve our narrative for the next sessions, for instance, by adding the explanation of the line chart again after the interaction with the time slicer.

7. Discussion

In this section, we describe the extent to which our process model can be used for informing and evaluating onboarding techniques and how our model can, in turn, be evaluated by such a technique. We also discuss future strategies for evaluating the process model. Finally, we share the lessons learned by highlighting some of these techniques during the creation of our process model.

7.1. Evaluation

As described in Section 6.1, our model can only inform the authoring and evaluation process to the extent of providing a scaffold for onboarding a user. It is up to the onboarding author to make the final choices required for onboarding, weighing additional factors that
are beyond the scope of this process model. Some of these factors are the author’s presentation, engagement and language skills, the amount of user interaction planned, and the skill set of the user, which all have a high impact on the onboarding process.

Failing to consider these factors and making incorrect choices for the questions raised by our model, can lead to an ineffective onboarding. Therefore, evaluating a specific onboarding—even if it was created based on our model—does not directly evaluate the model but rather the multitude of choices made by the onboarding author along the way.

One way to evaluate the process model is to test it against new onboarding strategies (as done in Section 5) to determine where the process model fails to describe them. Although we covered diverse cases in Section 5, we acknowledge that the proof of generalizability is limited and can only be solidified by future applications of the model. Therefore, as a next step, we plan to inform and evaluate future onboarding with our collaborators using our process model to establish its validity.

7.2. Lessons Learned

We share the lessons learned during the course of our work. We briefly discussed the main reason behind discarding most of the initial versions of our process model in Section 4. Here, we describe additional complications arising from the interplay among different user roles as indicated in Figure 9. Even without unpacking the usage and the onboarding loops, making all roles (apart from the user) implicit in the model. Rather than focusing on who updates the onboarding content, we focused on the actions that trigger updates.

Figure 9: Initial process model showing four roles involved in the dashboard onboarding pipeline.

After multiple iterations of process models, we also realized two additional advantages of this viewpoint. First, it allowed us to draw parallels between the interactively updated views of the dashboard with the dynamically updated content of the onboarding, leading to a symmetric process model. Second, it naturally led to the user being placed at the center of the usage and onboarding process as the sole explicit role.

8. Conclusion

Supporting users in their dashboard literacy is a major challenge [SCB+19]. We have presented a process model for dashboard onboarding that formalizes and unifies various onboarding strategies. We, therefore, introduced the usage loop, where users interact with the dashboard which results in the manipulation of either the underlying data model or the dashboard components. Alongside the dashboard usage loop, we introduced the onboarding loop, where the user manipulates onboarding artifacts or the underlying content through interactions. We demonstrate the generalizability of our model by applying it to four real-world examples (video, textual explanation, programmed guided tour, and interactive onboarding with a human presenter) and proposed a hypothetical usage scenario using an AI-based onboarding that sought to solve the flaw in interactive onboarding with a human presenter. Furthermore, we provided actionable advice for developing new dashboard onboarding systems systematically and efficiently. We are confident that our process model will help to create a structured dashboard onboarding and to improve user understanding of dashboards.

Acknowledgements

This work was supported in part by the FFG (881844). Pro2Future is funded within the Austrian COMET (Competence Centers for Excellent Technologies) Program under the auspices of the Austrian Federal Ministry for Climate Action, Environment, Energy, Mobility, Innovation and Technology, the Austrian Federal Ministry for Digital and Economic Affairs and of the States of Upper Austria and Styria. COMET is managed by the Austrian Research Promotion Agency FFG. Additional support was granted by the State of Upper Austria and the Austrian Federal Ministry of Education, Science and Research via the LIT – Linz Institute of Technology (LIT-2019-7-SEE-117), and by the State of Upper Austria (Human-Interpretable Machine Learning). This paper was written in collaboration with the VRVis Competence Center. VRVis is funded by BMVIT, BMWFW, Styria, SFG and Vienna Business Agency in the scope of COMET (854174).