Use of Tools: UX Principles for Interactive Narrative Authoring Tools

DANIEL GREEN, Bournemouth University, Faculty of Science and Technology, UK
CHARLIE HARGOOD, Bournemouth University, Faculty of Science and Technology, UK
FRED CHARLES, Bournemouth University, Faculty of Science and Technology, UK

The technology supporting Interactive Digital Narrative (IDN) is of particular significance to cultural heritage research. IDN technology provides a means of engagement in cultural heritage sites, a medium for culturally significant stories, and culturally significant story-centric games. While previous work in this space has numerous examples of user experience (UX) evaluations of the interactive narrative works themselves, there is significantly less in terms of evaluation of technology for authoring IDN, creating a UX research space in this area that is focused on audience and not authors. We propose to balance this focus by considering the UX of authoring tools more closely. In this work, we undertake a review of the state of the art of authoring tools for IDN such as story-centric games, and report on a rigorous UX evaluation of representative technologies (n=21). We also address the challenges of UX research for these tools through an original evaluation methodology where authors complete a story composed of representative story features. Our study leads us to conclude 7 UX principles for IDN authoring tools that both explore how authors use tools to create story-focused games, and how the interface for these tools impacts the creative process.

CCS Concepts: • **Human-centered computing** → **HCI theory, concepts and models**; *Empirical studies in interaction design*; • **Software and its engineering** → *Interactive games*; • **Applied computing** → Media arts.

Additional Key Words and Phrases: Interactive Narrative, Authoring Tools, User Experience

ACM Reference Format:
Daniel Green, Charlie Hargood, and Fred Charles. 2021. Use of Tools: UX Principles for Interactive Narrative Authoring Tools. *ACM J. Comput. Cult. Herit.* 1, 1, Article 1 (January 2021), 26 pages. https://doi.org/10.1145/3458769

1 INTRODUCTION

Interactive Digital Narrative (IDN), including story-centric games, have become a part of our collective cultural heritage as a means to tell culturally significant stories (such as work with the Mostar bridge [52] and similar work led by Katifori [28] and Vrettakis [56]), as a tool for engagement in culturally significant locations (as seen in numerous digital locative narrative works [6, 55]), and as culturally significant artistic works in their own right (from classic works of hypertext fiction such as Joyce's 'afternoon, a story'[1], to modern BAFTA celebrated games such as Kojima's 'Death Stranding'[2]). Interactive entertainment provides a medium with an immersive potential unlike traditional media [14], and digital technology provides new ways for an audience to interact with cultural heritage such as virtual or augmented reality [25, 32, 33, 46, 51], or through contextual awareness

[1]Michael Joyce, 'afternoon, a story', published by Eastgate, 1987
[2]Hideo Kojima, Death Stranding, published by SIE, 2019

Authors' addresses: Daniel Green, Bournemouth University, Faculty of Science and Technology, UK, dgreen@bournemouth.ac.uk; Charlie Hargood, Bournemouth University, Faculty of Science and Technology, UK, chargood@bournemouth.ac.uk; Fred Charles, Bournemouth University, Faculty of Science and Technology, UK, fcharles@bournemouth.ac.uk.

XXXX-XXXX/2021/1-ART1 $15.00
https://doi.org/10.1145/3458769

of location [23, 38, 42, 45, 54]. We see evidence of the importance of these systems in recent work in our field — in their recent survey of the state of the art in user experience (UX) research in cultural heritage technology, Konstantakis et al. describe IDN as "widely recognized as a powerful means for augmenting the engagement of people and achieving the goal of communicating knowledge," [31] pointing to landmark IDN cultural heritage applications [37, 44].

Despite this potential, however, authorship remains a notable challenge in this medium. The technical skills required to develop game-based stories can act as a gatekeeper preventing creatives from using the medium [41] or being prohibitively expensive to resource in a challenging financial climate for cultural heritage. The accessibility of this medium can be considered a key concern for this field given the already identified concerns and challenges regarding the adoption of new technologies in cultural heritage [27]. The research community's collective solution to accessibility for creatives in this space has been authoring tools — software aiming to enable creatives to design and build interactive narratives. While some of the research in this space inherits design principles from programming IDEs, interactive narrative authoring tools often aspire to the added challenge of providing access to less technical or at least non-programmer users — authors who might use such a tool to create interactive stories with a creative or scholarly, but not necessarily technical background [40]. Direct parallels to this can be seen in the research on End-User Development [36] authoring tools in the cultural heritage space, where authors are typically curators, guides, or other professionals that may lack technical abilities to operate more complex systems [3, 15, 16, 23, 38, 49]. This demands careful consideration of the interface design and UX of these authoring tools. However, while there are established UX principles/heuristics [12, 13, 34] and evaluations [2, 47, 48] for the games or interactive experiences themselves, comparatively little of this work focuses on authoring tools, the emphasis being firmly on the games and cultural experiences created. The recent state of the art reviews by Konstantakis et al. [31] and Bekele et al. [4] reflect this where the research works reviewed are evaluations of cultural heritage experiences, games, and IDNs themselves, never the tools required to created these valuable experiences. This has lead to a UX focus in this space on the audience rather than authors. The few works reviewing the state of the art in IDN authoring technology that do exist are concerned more with classification and identification than critical evaluation [53]. While these classifications are valuable they lack the careful exploration of a UX evaluation, telling us more what exists rather than what is needed in an effective authoring tool. It is possible that limited evaluation work on authoring tools is due to the fact that evaluating interactive narrative tools presents unique challenges for UX research, in a field where the established methodologies often employ 30-60 minute user study tasks [18], and where writing a meaningful story takes significantly longer. This can make use of the tool challenging to effectively evaluate with existing techniques, and demanding a new approach to UX evaluation in this area.

In this paper, we set out our own evaluation of the state of the art in this critical area for digital cultural heritage, a novel methodology for evaluation of these applications, and the key principles that may be derived for future authoring tool design and evaluation. This work will seek to establish these key principles and answer our research question: *what is the impact on authoring workflow for different user interface paradigms for interactive narrative authoring tools*? Here, 'workflow' refers to the method by which authors write in terms of frequency of actions taken, focus on particular actions, and the ordered flow of the actions. This research question attempts to build a better understanding of what creative impact a given authoring tool has with respect to its exercised interface paradigms upon the authoring experience for creation of interactive narratives.

2 UX, AUTHORING TOOLS, AND STUDY PREPARATION

When designing the interface of any system, whether it be an application, a website, or other interfaces, it is important to identify and understand the way in which design decisions will impact a user's experience with the system. If interacting with a system is unclear or difficult, then cognitive friction [9] will rise, which can bring about frustration and result in a negative user experience. This is especially important for authoring systems where we want to avoid distracting users from the creative process of authoring an interactive narrative. General design heuristics for the design of and interaction with interfaces have existed for quite some time [43]. These principles are ideal for high-level design but do not aid us in understanding the impact of these design decisions in domain-specific processes, such as the creative workflow of authoring for an interactive story. Therefore, we must go beyond these best practices to identify the impact of various design decisions within the context of authoring interactive narratives. By determining the impact upon creative workflow that different design decisions have, we can better inform designs of new authoring systems.

2.1 Authoring Tools

As mentioned above evaluating the UX of authoring tools presents unique challenges in that the tasks typically undertaken by users are as long and varied as any creative work. Furthermore, there are a variety of technologies available in this space and to begin to study the state of the art in this area demands we identify a representative selection of the existing approaches. Preceding this work we surveyed the current state of the art in interactive narrative authoring tools [20]. In total, 29 individual authoring tools were included in our survey. 14 of the tools were sourced from academically published literature. Four are developed and sold as commercial products. The remaining 11 come from other non-commercial, non-academic sources such as open-source or otherwise free projects. This distinction is important as the purpose of these tools can differ based on their origin; commercial products are developed with a different goal than research projects, which can impact the focus and quality of these tools both in terms of features and user experience.

2.1.1 Availability. When evaluating the included authoring tools, it is important to determine their availability. In this context, availability refers not only to the accessibility of the end product (i.e., executable binary, web service, etc.) but also their online presence and whether their source code is accessible in the case of products that are licensed accordingly. Within this definition, we must also account for systems that used to be available but have since succumbed to fates such as presence of dead links, terminated websites, and software entropy[3].

The availability of a tool directly impacts its ability to be adopted by an authoring community and its ability to be further developed or otherwise used for research. Systems that have long become dormant can be devalued relative to their initial contribution, as they are unlikely to be adopted in the long term by authoring communities unless kept at least functional. *Inform*[4], for example, has been developed and used since 1993, but due to its strong online presence and dedicated community-driven development, it still remains a strong contender among the interactive fiction authoring community even today.

Online presence is an area that academic authoring tools struggle to maintain. Of the 14 selected tools from an academic background, only six ever had a dedicated website, of which only four still exist today and are seldom, if ever updated. All other tools (except *HyperCard*[5], which was discontinued in 1998) have an active online presence either through dedicated pages or a repository in the case of open-source projects. It is not uncommon to encounter research projects that are hosted on a specific academic's personal page. These websites are susceptible to becoming invalid, such as when the researcher leaves the institute and the personal pages are terminated. This, and other temporary websites often result in unreachable links and are not a suitable substitute for a long-term dedicated project page where tools can be publicized. To illustrate this, an article of the academic authoring tool *Emo-Emma* [8] provides links to academic papers and binaries, but all of the links reside on a university staff page that no longer exists, making it difficult, if not impossible to source the original contributions.

[3] Also known as *bit rot* as defined by *The Jargon File v4.4.8* (http://catb.org/jargon/index.html)
[4] *Inform* by Graham Nelson, 1993. Available at http://inform7.com as of 13 November 2020.
[5] *HyperCard* by Apple Computer, Inc., 1987.

Another area that the selected academic tools struggle with is the distribution of binaries (or other runtime) and source code, where applicable. Some projects are understandably protected intellectually, but those that are not should attempt to share their work to better maximize their chance of adoption by authoring communities, and provide opportunity for further research to be conducted. Of the 14 academic tools, only five ever offered binaries at some point, with only one remaining publicly available today (*StoryPlaces* [24], which at the time of writing offers both a web service and source code). Sometimes software is made available upon request. *StoryTec* [17], for instance, used to provide a software request form online but has since removed it in 2015. The *ASAPS* [30] software has a request procedure that we had followed in mid-2018, but were unsuccessful in obtaining the software as no response was provided with adequate time given. This highlights the need for care to be taken to ensure that if software is intended for public use, that it is made, and remains, easily available, otherwise any traction gained could be rapidly negated. The lack of availability in the academic circle has serious research implications for the interactive fiction and authoring research communities, as it prevents the reproduction of any experimental results and hinders their study so that we might incrementally improve and iterate upon existing work, or form a greater understanding of existing works. While some conclusions can be drawn from documenting articles or publications, this is not a suitable replacement for interacting with the software itself.

For the authoring tools that were included in our survey, 15 out of 29 were available for public use. Both the tools sold as commercial products and those from other non-academic sources were wholly available, contributing 14 of the available tools, with only one of the academic tools being available for use. This further demonstrates the trouble of availability within the academic space.

2.1.2 Clustering Tools. To identify a representative selection of the broad tools available for study, we can identify clusters of tools based on their features and then pick representatives from these clusters. As the focus of this investigation is the UX of these tools, our features come from both UX and systematic affordances of applications rather than the capability of their underlying narrative models. A complete listing of the features used can be found in our previous work [20].

Each tool was systematically reviewed through a combination of actual usage and a review of relevant documentation and research papers to identify which features it related to. It should be noted that while every effort was made to retrieve and use the tools, in some cases, as discussed above, they were not available. In the case of a tool not being available to use, features were inferred from research papers and other sources like project pages or presentations. If a given feature for a tool could not be ascertained, the feature was assumed not to be supported instead of imputing to avoid declaration of non-existent features.

Clustering was done using the **R** language and related third-party packages, such as FactoMineR [35]. As the data has individuals described by purely categorical descriptors (i.e., only enumeration and binary columns), Multiple Correspondence Analysis (MCA) was used as a preprocessing stage, followed by Hierarchical Clustering on Principal Components (HCPC) to determine clustering of the tools based on the features, as described in the `FactoMineR` documentation. HCPC generates a tree structure that can be cut to determine the number of clusters. By default, HCPC requires the number of clusters to be specified by the user. However, it is possible to request HCPC to suggest a best-case number of clusters calculated by inertia gain[6] as the number of clusters increases. In this case, we used the number of clusters that was suggested by the algorithm, which was four. A planar projection of the endpoints of the 3D tree generated by the HCPC algorithm can be seen in Figure 1 (right). This figure visualizes the clusters by color. Note that distance in two dimensions does not necessarily correlate to similarity as the points are the projected result of a branching hierarchical tree in three dimensions. However, the dendrogram depicted in Figure 1 (left) is a 2D representation of such a tree and we can use the branches of this to infer similarity and divergence. The dendrogram's height value (y-axis) can be used to determine the potential result of increasing or decreasing the requested cluster count. For instance, increasing to five clusters would split

[6]Inertia is a measure of variation used to determine when it is no longer beneficial to add clusters.

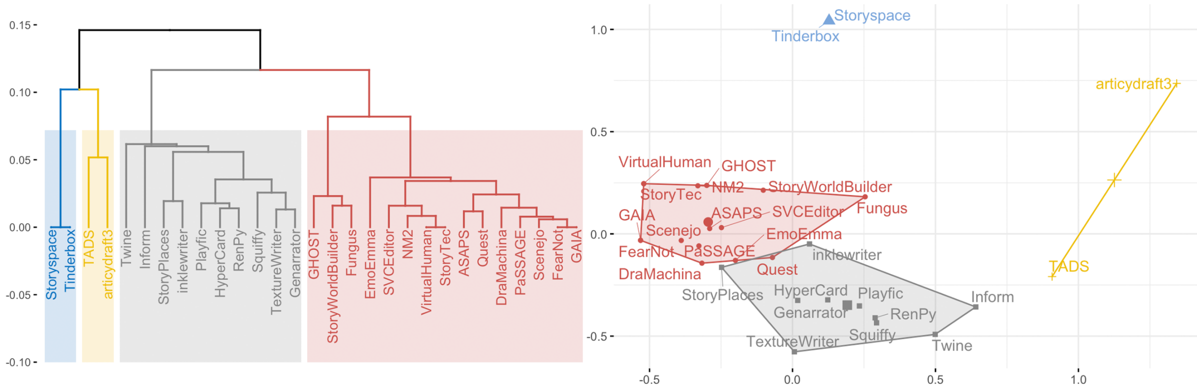

Fig. 1. (Left) Dendrogram of HCPC result clusters 1 (Red), 2 (Blue), 3 (Green), 4 (Yellow). (Right) Planar projection of the 3D HCPC clustering algorithm tree.

the rightmost group unevenly in two. However, as the inertia gain generated by HCPC suggested four clusters as optimal, this was not changed.

2.1.3 Representative tools selected. For our study, *Quest*[7], *Inform 7* (*Inform*), and *articy:draft 3* (*Articy*)[8] were chosen. The first two of these tools come from the largest clusters in the analysis, and the third from one of the smaller clusters. Each of these programs presents variety in their UX and exercised UI paradigms, which we have previously enumerated for IDN authoring tools [19]. Figure 2 shows the three authoring tools in context. *Quest* makes heavy use of multifaceted system controls such as text boxes and combo boxes, *Inform* hosts a natural language scripting system within an augmented text editor, and *Articy* primarily uses a dynamic node graph. This variation ensures that as many of the primary UI paradigms can be tested, rather than choosing, for instance, two systems with similar interaction paradigms from different clusters. From their individual clusters, *Articy* was chosen due to its popularity within creative industries[9] and *Inform* was chosen due to its popularity with interactive fiction communities[10]. *Quest* was chosen as within its cluster only itself and *Fungus*[11] were available for use, and *Quest* presents a much more multifaceted-centric experience, which was the only paradigm not yet covered. *Fungus*, on the other hand, does have multifaceted controls, but also requires a graph and scripting, both of which were already covered by the other selected tools.

The leftmost cluster in the dendrogram in Figure 1 contains two authoring tools from the same publisher. These tools are similar (one is for general hypertext editing and the other is more specialized for IDN) and both primarily exercise a graph-based editing experience. This type of interface paradigm was already covered by the inclusion of *Articy*, and given this is a narrow cluster representing 2 very similar tools, consequently neither of the tools from this cluster were selected for the study.

2.1.4 Representation of the Selection. The broad population of authoring tools [20, 53] demands that in studying them we must attempt to take a representative sample. The process we describe above demonstrates our means of taking a reasonable representative sample of authoring tools which we in turn have analyzed to identify three

[7] *Quest* by Alex Warren, Community, 1998. http://textadventures.co.uk/quest
[8] *articy:draft 3* by articy Software GmbH & Co. KG, 2017. https://articy.com
[9] Articy boasts several prominent game studios among its users including THQ Nordic and CD Projekt Red.
[10] As of writing, IFDB (www.ifdb.tads.org) shows 4405 stories written using a version of *Inform*.
[11] *Fungus* by Chris Gregan & Steve Halliwell, 2015. https://fungusgames.com

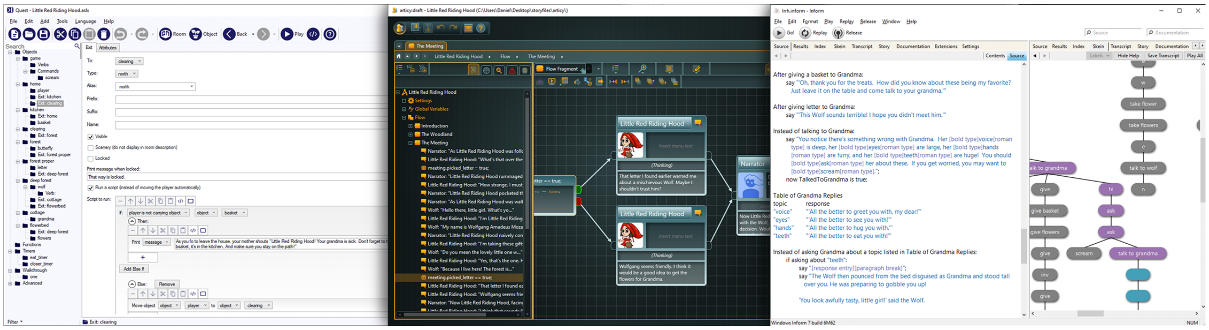

Fig. 2. The three authoring tools included in the study. *Quest* (left) primarily uses a multifaceted control-based interface, *Articy* (center) primarily uses a dynamic node graph, and *Inform* (right) primarily uses an augmented text editor.

representative tools for our study. Inevitably, our resulting selection does not include all tools, but we have taken steps to ensure the greater population are still broadly represented within our selection. This greater population includes a number of authoring tools from the cultural heritage sector such as the *CHESS* authoring tool used by Roussou et al. [49], the *EMOTIVE Authoring Tool* [29] (*EAT*), *Cicero Designer* [16], *TaggingCreaditor* [54], and *ARIS* [25], among others, which, while not explicitly surveyed, remain represented in their UX paradigms here. For example, the *CHESS* authoring tool chiefly relies upon a dynamic graph to construct interactive branching narratives, and *EAT*'s *Visual Scenario Editor* uses a node-based visual scripting system to create narrative experiences, which can also be found as the primary editing method within *Twine*[12], *Storyspace* [5], and *Articy* from our selection. *Cicero Designer*, *TaggingCreaditor*, and *ARIS*, on the other hand, each make extensive use of multifaceted user controls for configuration and editing of the produced narratives, with parallels to these interfaces being found in *Quest*, *Fungus*, and *inklewriter*[13] within our authoring tool selection. Our authoring tool selection also covers a wide range of tool contexts from games authoring (*Articy*) to traditional hypertext narrative (*Storyspace*) to locative systems frequently used to deliver narratives for cultural heritage (*StoryPlaces* [24]). It also covers a variety of sources ranging from paid proprietary tools like *Articy* to free community tools like *Inform* to academic research projects such as *Emo-Emma* [8].

Consequently, we are confident that our broad selection of authoring tools, and those ultimately chosen from the clusters for use within the study, are representative of the wider field of related authoring tools, including those found within varying disciplines of cultural heritage, even though we were unable to individually analyze each individual tool.

2.2 Representative Story Features

Since our study focuses on interactive/game narrative authoring tools, any UX experiment must make use of a task for participants that mimics contemporary game writing. Since there are countless variations within the vast corpus of video games writing, we likewise need to use a representative selection of writing challenges and features for this task that mimics the genuine act of creating interactive narratives. Towards this, feature analysis of a set of story-focused games was done to determine a listing of commonly employed storytelling features. These features were derived from common game design theory literature such as Mason's model of choice [39], Booth's narrator classification [7], Jenkins' refinement of environmental storytelling [26], our own work on Discoverable Narrative [21], and other tropes found in the medium as described in other literature [1, 50]. The

[12] *Twine* by Chris Klimas, Community, 2009. https://twinery.org
[13] *inklewriter* by inkle Ltd., 2013. https://inklewriter.com

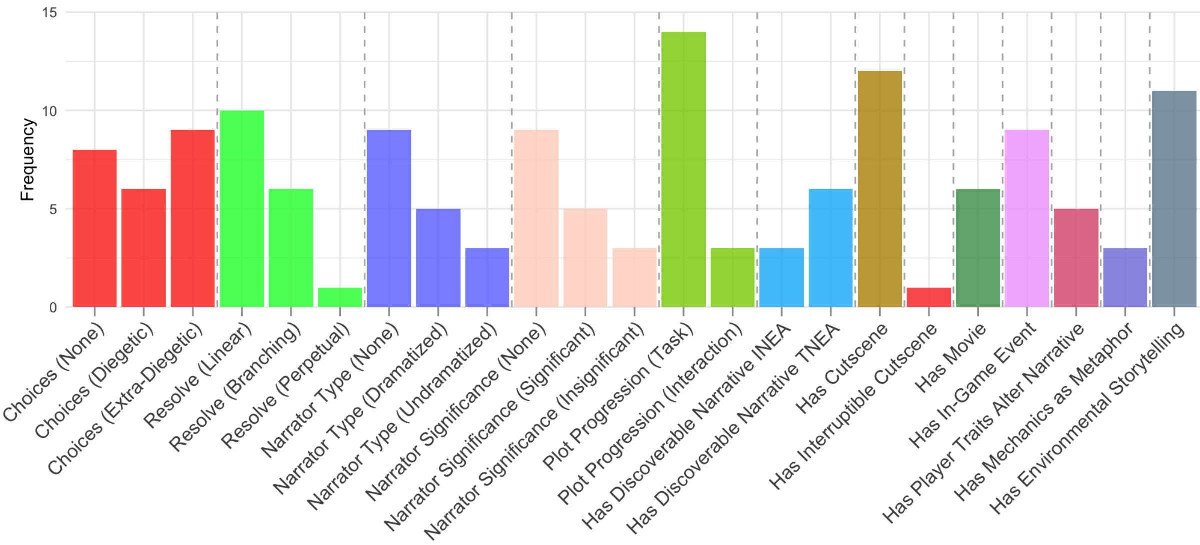

Fig. 3. Game story features frequency analysis. More details can be found in Appendix A.1.

frequency of occurrence of these features across the selected games was then used to influence task design in the story templates (discussed in §2.3).

In total, 17 video games were included in the analysis from a broad range of genres, typically being selected based on their prominence as a narrative experience. Each game was played through and analyzed to determine high-level narrative features employed by the authors. **R** was then used to conduct a frequency analysis of the individual features over all games. Results can be seen in Figure 3. It is clear from this graph that some features, such as perpetual endings and interactive cutscenes, are insignificant within the corpus analyzed, whereas others, such as standard cutscenes and environmental storytelling, are prominent. A complete listing of the features and their detailed descriptions can be found in Appendix A.1.

2.3 Evaluation Story Template

When designing the methodology, we initially considered allowing participants to write a story of their choosing as to not limit creativity. However, as previously discussed, this is challenging to study within our domain given the time it takes to write a full interactive narrative, and the potential fluctuations in participant creative inspiration. Taking into account our demographic, we elected for our methodology to instead have participants finish a partially completed story. Participants would be presented with a partly written story and asked to complete it, and its missing features, while also giving them some limited creative freedom in how they did this. This way, the context under which participants were acting was relatively consistent, there was sufficient time to pragmatically evaluate the UX for each participant, they were undertaking to write for a story representative of interactive narrative, and the chance of 'writers block' coming from a blank page (which might waste study time) was reduced.

The common game storytelling features identified earlier were formulated into a planned interactive version of Little Red Riding Hood (LRRH). This particular fairytale was chosen due to its ubiquity and international presence to reduce the chance of a participant being unfamiliar with the story's arc. It is also a commonly used story within the IDN research community. The inclusion of a partially completed LRRH adaptation serves as a means for participants to interact with the authoring tools in a context that is relevant and populated with recognizable content. The story was broken up into an introduction and three distinct chapters. In the introduction, LRRH is set on her way by her mother. In the first chapter, LRRH makes her way through the woodland; this happens prior to her encounter with the wolf. In the second chapter, LRRH is confronted by the wolf and then continues to her grandmother's cottage. In the third chapter, the wolf had already arrived and attempts to eat LRRH.

A story template implementing the LRRH script was then created for each of the chosen authoring programs. This template amounted to an adaptation of the LRRH story detailed above in each authoring tool in the medium appropriate to that tool. Certain liberties were taken with the story as demanded by the medium — for example adapting the story for the adventure game, room based, style of *Quest* and *Inform* resulted in a quite different interactive version of LRRH to that of the dialogue trees favoured by the narrative model in *Articy*. However, each template still followed the same overall chapter structure, and still made use of the representative narrative features described above. This template would later be edited to remove a section which participants would be asked to complete as the basis for the study.

2.4 Training Videos and Documentation

Short annotated training videos were prepared for participants to watch prior to their use of the tools to ensure a baseline level of knowledge for all participants. The three training videos introduced the interfaces and their essential functionality by building a fictitious story that exhibited many of the features of the LRRH story that they would be using in practice. Care was taken to ensure that the tutorial videos differed from and did not include aspects of the LRRH story, but instead provided the participants with an understanding of the tools required to complete the study.

A documentation webpage was also created for participants, which was available to them to read prior to and throughout the experiment. The page contained tutorial text explaining the procedure of the experiment as well as instructions on what they are expected to do. As each chapter in the LRRH story was based upon a set of typical authoring tasks in games writing derived from the story features analysis, an overview of the LRRH story chapters broken down by the respective tasks that they are made up of was presented in this document also. This consisted, for each task, of the name, a general description of the task, and a grounded example of how it was implemented into the story template. The remainder of the page contained the LRRH script segmented by chapter and color-coded by character. Participants were informed that they could use this document to copy from in the event that they want to follow the given script.

3 USER STUDY PROTOCOL

To answer the proposed research questions for this experiment, participants must use interactive narrative authoring tools to engage in a typical authoring scenario associated with games writing. The tools included in the experiment — *Quest*, *Inform*, and *Articy* — were chosen from the clusters discussed earlier as representatives of the broad features and interface paradigms found in the complete selection of considered authoring tools.

The target demographic for this experiment is students and professionals of digital narrative who have an interest in creative writing, particularly for the application of video games writing. Participants are not expected to be experts in writing, but it is expected that they are competent users of computers. Both student and academic participants were sourced from related courses and disciplines from Bournemouth University, Southampton University, and one from Arts University Bournemouth. A total of 21 participants were involved in the experiment, consisting of 14 students and seven academics. Our target demographic is closely related to other cultural heritage studies involving End-User Development [36] authoring tools. While the role that participants fulfil may be specific to a given discipline (such as curators for museum experiences [16, 49], guides for cultural heritage sites [3], or in our case interactive narrative storytellers), similarities are maintained in that the target audience for both the study and sometimes the product are those with an interest in delivering a narrative experience but typically lack the technical knowledge to operate complex systems. An incentive was offered to participants to encourage participation and reward their contribution. Any participants that completed the experiment were optionally entered into a prize draw with a chance to win one of three £30 Amazon vouchers. The winners were randomly selected after all data was gathered and processed; participant performance was not taken into account.

As part of the preparation, each participant was assigned one of the three tools and were shown the corresponding tutorial video for their assigned authoring tool. Participants were not timed when watching the video so that they could pause and rewind according to their preferences. This ensures that all participants have at least a base level of knowledge required to complete the study. Accompanying project files from the training videos were optionally given for participants to explore before the experiment proper, but none wished to do so. The training video was also allowed to be referenced again during the main part of the experiment and counted as accessing a source of documentation.

The starting template that participants receive implements the script as earlier discussed. However, the second chapter, where LRRH meets the wolf, has been removed and replaced with empty placeholder content. Instruction is given in the edited templates for what (but not how) the original unedited script had done to fulfill this now missing section to serve as guidance. The challenge for participants then becomes filling in the missing content between the first and third chapters. The second chapter was chosen as it includes nonlinear pathways, dialogue, and quests, which provides enough variation for participants to be creative. The first chapter is comparatively linear, which may have produced less interesting results, and the third chapter involves complexities that may have been difficult for participants (especially those new to the software) to implement given time limitations.

3.1 Participant Flow

A single session for one participant takes around one hour in total to complete. Each participant is randomly assigned one of the three tools. Due to uncertainty with potential participant counts, it was decided that simply cycling through the tools will provide the most even distribution, as providing the participant count was a multiple of three, all tools would be evenly chosen. This is in comparison to a technique such as block randomization, which for a larger sample size would have been the preferred option. Before the experiment begins, participants are given an information sheet that describes the purpose of the experiment, the expectations, incentives, and so on. The participant can ask any questions about the procedure here. To begin, the participant is given the documentation page as described in §2.4. They are given as much time as desired to read the information about the experiment, familiarize themselves with the listing of tasks, and to ask any questions prior to their authoring session starting. The participant then watches, at their leisure, their corresponding training video and is offered the opportunity to explore the respective sample project built in the training video. Following the training, the participant is then given a hard limit of 30 minutes to complete the missing section of the template story in their assigned authoring tool. Participants are allowed to finish early if they consider their work complete. Participants are also allowed to ask questions during their authoring session. While participants are encouraged to link the first and third chapters together following the chapter's tasks, they were informed that these tasks are guidelines and that they have complete creative freedom as long as it remains within context of the story. This decision was made rather than having a strict set of tasks that must be followed to the letter to encourage a more natural authoring environment. It also provides structure for those that may struggle to come up with ideas under test conditions. Finally, the participant is interviewed after they have completed their authoring session. Each participant session was moderated sequentially by the same individual researcher, including preparation, running the study proper, taking detailed notes during the study, and conducting the interview.

3.2 Data Gathering

Data was gathered from multiple sources to maximize output of the experiment. Each participant was assigned a unique identifier which was used to collate all data gathered about them during the experiment and served as a way of retaining the grouping once anonymization had taken place.

While participants were implementing their story, observational notes were taken, focusing primarily on building a profile of the participant to aid in the post-experiment interviews. This included highlighting moments of obvious struggle and frustration, as well as periods of adept use of the authoring tools.

The screen of participants was also recorded along with audio capture. This data was gathered to allow for post-experiment analysis of the actions and usage of the authoring tools. Participants were also encouraged to speak aloud during their authoring process as to give more insight into their decisions. Unfortunately, few participants did this as they felt uncomfortable doing so.

Following the experiment, the participant filled in a short demographic survey which served as a way of grouping participants in the analysis stage. A complete listing of the questions can be found in Appendix A.2. After the demographics questionnaire, an interview was conducted with each participant to determine their experience and opinions with the authoring tool. A complete listing of the questions can be found in Appendix A.3. The questions were firstly asked in order, and then any significant points from the observational notes were raised with the participant, resulting in richer information for each question where points were available. The audio for this interview was recorded for later transcription.

Information collected about participants was kept in regulation with Bournemouth University's Data Protection Legislation. Data was anonymized and transcripts were written as soon as all participants had been through the experiment process. Participants who entered the incentive prize draw had their emails stored privately and were removed once the draw had taken place and the winners had been contacted.

3.3 Acknowledged Limitations and Problems

While efforts were taken to ensure that this experiment was fair, it is still important to make note of limitations and potential biases of the methodology.

Authoring tools for the experiment were chosen as representatives of the wider selection, best maximizing the difference in user interface paradigms. All of the tools represent key different approaches to enabling the authorship of interactive narrative but also share some aspects in common. For example, *Quest* and *Inform*, while from an authoring perspective provide vastly differing experiences, both use a room-centric "adventure game" style model of narrative, whereas *Articy* does not. This means that the tropes of adventure games, such as the use of 'rooms' and command based interactions, will be more prominent in two of the three softwares. It is therefore expected that while *Quest* and *Inform* have different authoring experiences, they are possibly more similar in overall writing style to each other than they are to *Articy*.

The original eighth participant was removed from the data and replaced with another using the same tool. With the original participant, there were misunderstandings due to a substantial language barrier (the participant was not a native English speaker) which also prevented accurate gathering of verbal data.

4 DATA ANALYSIS

Of the 21 participants, all but 3 participants (one student, two academics) had no previous knowledge of their assigned tool, with one academic reporting moderate knowledge and the other two reporting little knowledge. Most participants had little to no knowledge of similar tools, with three reporting moderate knowledge (one student, two academics) and two reporting plenty of knowledge (one student, one academic). Five academics reported moderate knowledge of interactive digital narrative, with the remaining two declaring plenty of knowledge and little knowledge respectively. Students were more varied with five reporting little knowledge, four reporting none, three reporting moderate, and two reporting plenty. All but three participants thought that the training video was satisfactory. Those that didn't expressed that they wanted a longer, more detailed training session. However, the video presented to them was condensed and limited in the interest of participant time and instead served as an introduction rather than a course. Only five people considered their own story complete. There is little variation of this between tools, other than *Inform* having one extra incomplete story versus the other two which each had five. Twelve of the participants had a functioning story where 'functioning' refers to being able to play through the story from start to end without error, as intended by the author. *Quest* has one of seven that didn't work due to incorrectly configured objects, which suggests that it is relatively easy to make a functioning story in this tool. *Inform* had a mixed rate of success with two compiling properly, three compiling but having syntactical errors resulting in incorrect behavior, and two that didn't compile at all due to errors. *Articy* had three functional and four nonfunctional stories, with all four having some form of missing connections. Of those four, two had missing connections at the beginning and end of the story, and the other two are generally unfinished along with syntactical errors.

4.1 Workflow

Each of the screen recordings were manually annotated with timestamps and indicators of a given action that a participant did. Once recorded into a timeline, both frequencies of these actions and their ordering could be determined for analysis. The actions themselves were enumerated based on the observed behavior of participants rather than being defined beforehand. A single action represents a vignette of the larger authoring experience. It was decided to only include the occurrence of a given action rather than to record its duration due to variability and noise when doing so (for instance, a participant stopping to think about what to write shouldn't extend the time of a previously unfinished action, and this cannot be easily determined). Each action has an assigned letter, a description, and where applicable a program-specific instruction that encapsulated what activity should be considered as a particular action taking place. For example: "World Building (W) — Present text to the player that describes and conveys the state of the world". A complete listing of the actions and their descriptions can be found in Appendix A.4.

Figure 4 shows a stacked bar chart of the individual action frequencies for all participants colored by authoring tool. This shows the total occurrence of actions across the entire experiment as well as the distribution of actions between the tools. It is possible to determine patterns in the authoring workflow by combining a single participant's timeline into a single consecutive string and using a sliding window to generate ngrams. For instance, actions `A` then `B` then `C` becomes `ABC`. These combined strings are referred to as flows. When all participants of a given tool have their flow ngrams combined, it is then possible to determine any repetitive patterns, their frequencies, and how many participants contributed to them. This means that for any given ngram of size n, we can determine the patterns that exist per tool, and we can also cross-reference these patterns to further determine what common patterns exist between the tools. An important point to consider with ngrams is their use of a sliding window which makes them sensitive to long sets of repeating characters. For example, if we have the string *'tttttt'*, an n of 2 will result in five sets of *'tt'*, when actually the full string is a truer representation of the pattern. This can artificially inflate the reported number of occurrences. However, a high number still signifies that this feature is important, just perhaps not as important as it appears. A solution to this is to increase n and see if similar patterns continue to emerge. In the case of this analysis, n was run from 2 through 8. *Inform* had no occurring patterns from 4 onwards and *Quest* from 5. Any matching patterns between tools ceased to exist after 2. Noting how many participants contributed to a pattern is important, as it is possible to have a high number of occurrences that are split between a small number of participants, or vice versa with a small number of occurrences split between many participants; these both mean different things. For this analysis, a threshold was set so that only patterns that occurred with three or greater participants were displayed.

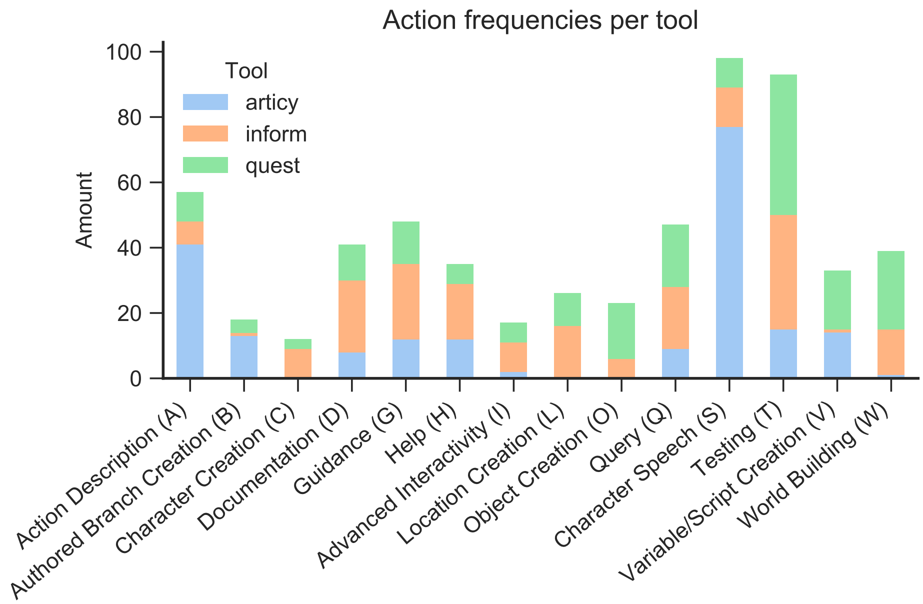

Fig. 4. Action frequencies for all participants.

To further clarify the likes, dislikes, and desires of participants with regard to the experience of the authoring tools, all interviews were transcribed and subject to an axial coding strategy. This helps to break down the qualitative data (i.e., the opinions of participants) into quantitative values that can be accumulated to determine popularity of thoughts. The questions in the interview focused on likes of the experience, dislikes of the experience, desired modifications of the software's existing feature set, and additions to the software's feature set. Axial coding categories were iteratively chosen as participants were processed, attempting to fit their comments into existing categories, expanding if necessary, Each category had a purpose (what was it demonstrating), an observation (what was the occurrence doing), a list of criteria for what consists as a valid entry, and a tally of both the total occurrences per tool and the total number of participants involved per tool. A listing of these categories and their criteria can be found in Appendix A.5. The same individual researcher that moderated each of the participant sessions was also responsible for the axial coding.

5 PRINCIPLES

After detailed analysis of the data gathered from the experiment, a set of guiding principles for consideration during design and development of authoring tools supporting interactive narrative were derived. These principles serve as indicators for the potential impact of a given design decision on the way in which authors work.

5.1 Principle of Metaphor Testing

Interfaces that use a visual metaphor to represent story structure and connectedness will result in less testing of non-complex stories.

All participants using *Articy* tested at least once, with the median being two times. One participant tested five times, but this was an exception likely due to their use of more advanced features. The visual flowchart-like interface serves as a metaphor for an abstraction of the underlying structure and connectedness of the story. This kind of visual design is something that is intuitive and familiar to most people. The visual nature of nodes displaying characters, text, and having clear connectivity can serve as a lightweight replacement for testing, as authors can visually see what would happen when playing without testing simply by following the chain. This was observed during the experiment, as participants would periodically scroll back through their story to remind themselves of the preceding content and structure before continuing, without explicitly testing. In this scenario, testing would then be required only when choices or other interactions that are dynamic in nature and cannot be inferred easily from connectivity occur, such as branching based on conditions or dynamically enabling connections.

In *Inform*, all participants tested at least once with the median being five. One participant tested 12 times, but this was an outlier. *Inform's* code is typed in, and the only way to test if something functions as expected is to test it in-game. There is no visual way to tell what is going to happen and when, and since the code is compiled and does not execute sequentially, it cannot be read step-by-step to see what may happen. The higher number of testing instances is therefore justified by the lack of a visual metaphor that supports lightweight testing.

Quest visually lies in the middle of the other two systems with its abstracted scripting system providing some indication of runtime behavior, but otherwise has little in the way of visualizing structure. Testing must be done to determine most of the behaviors. Two participants didn't test at all and the median was five with one participant testing 17 times, but this was an outlier. Given that a total of 18 scripts were created across participants and that the abstracted script editor should assist in understanding what will happen when played, it should be expected that less testing than *Inform* would take place. However, even removing outliers and considering that two people didn't test in *Quest* at all, the total number of testing sessions is still greater than in *Inform*. This highlights that this principle is tied to visual metaphors of structure and connectedness rather than ordered listings of actions.

5.2 Principle of Fast Track Testing

Letting users jump to any state of the story enables more rapid and focused testing sessions.

In *Articy*, testing is instigated by selecting a node and either using a toolbar button or context menu of that node to enter into testing mode. This will fast track the testing session to begin at this particular point. All participants used this feature when testing every time unless initially checking the whole story at the beginning of the experiment. Entering into the desired starting state appeared to be intuitive as not a single participant had to ask how to enter testing at a given node. *Articy* also features a Journey mode which lets authors record pathways chosen so that not only can the state be restored, but also the particular pathway repeated. With the scope of the stories created in this experiment, participants did not feel the need to use the Journey feature, and only one participant queried about its functionality at the end of their session out of curiosity.

When testing in *Inform*, authors can use the Skein feature which shows a hierarchy of all actions taken during previous test runs with visual branches indicating where paths differ. This is not for editing, but for debugging and restoring the state of a previous test. Five of the seven participants using *Inform* made use of Skein more frequently than testing from the beginning of the story. Five of the seven participants explicitly complimented Skein for its usefulness in restoring states in their interviews. *Inform* also supports a syntactical `TEST` command which has a similar function, but this was not explored by any participants.

In *Quest*, there is a Walkthrough mode akin to *Inform's* Skein, but it is not visual and serves purely as automation of user commands to get back to a particular state. A single Walkthrough was provided in the starting project that participants received, but nobody attempted to create their own. Three of the five participants that tested in *Quest* almost exclusively used the Walkthrough mode when testing, with their first round of testing being manual from the beginning to understand the story before authoring commenced. The other two did not seem aware or interested in Walkthroughs, even though it was explained to them and demonstrated in the training video. An additional feature unique to *Quest* is clickable links for interactivity during testing mode, which helps authors get back to a given state quicker and makes exploring easier, especially when commands are unknown. This was used extensively by all to speed up testing rather than typing in commands, with it being the primary method of interaction in all but two testing sessions across all of the participants.

These points across all tools highlight the importance of allowing the user to return to a given state of the story as quickly as possible. Without this, users must traverse from the beginning, which wastes time and runs the risk of losing focus.

5.3 Principle of Structure

Interfaces that use a visual metaphor to represent story structure and connectedness are more intuitive for management of an author's own story structure.

Articy has the most obvious visual mimicking of underlying story structure and connectedness with its flowchart-esque design composed of nodes that are connected with lines. All seven participants spoke positively about their experience with the graph system, and described it as neat, intuitive, fluid, and highlighted that it is good for non-technical users likely due to its inherent familiarity. It was also complimented for its organizational abilities that go beyond pure story structure into personal organizational aesthetics (e.g., the ability to move and position nodes as desired to create pseudo-collections), which is something that the other two programs lack.

In *Inform*, three people stated that they would prefer a dynamic graph akin to that found in *Articy* for structural creation and management. *Inform* has a static map that updates only upon successful compilation, which four participants used and two explicitly commented upon as being helpful in debugging connections. Only two users complimented the text-based methods for creation of world layouts in *Inform*. In *Quest*, three people requested the same dynamic graph feature for the same reasons, and only two complimented *Quest's* form-based approach to world creation and management. In contrast, nobody using *Articy* requested an alternative way of creating and managing structure.

5.4 Principle of Experimentation

Interfaces that use a visual metaphor to represent story structure and connectedness enable easier experimentation of structure and connectedness.

Articy having connections created and displayed visually mimics the underlying structure with a flowchart-like design, which is something that most, if not all authors will be familiar with functionally regardless of technical ability. This familiar node-link setup makes it easier to experiment with structural configurations, as objects and their connections can easily be created, destroyed, and tracked without much cognitive friction due to the elements being presented together in a familiar and intuitive format. It was observed during the sessions that all participants in *Articy* were much more willing to experiment by disconnecting and reconnecting elements. It is easy for participants to keep track of what is and is not connected due to the visual approach. On the other hand, participants using *Inform* were observed to experiment with structure less. This is likely due to the lack of visual coherence of the structure of the story, and with a lack of clarity as to what elements are being used and which are redundant largely due to the program being almost exclusively text based. For example, two participants using *Inform* had renamed the default *MyRoom* in favor of creating their own room, but due to the lack of visuals, did not rename all instances in the text, and therefore *MyRoom* was still created and their renamed room was unknowingly disconnected from the rest of the story. Similarly, *Quest* suffers from a lack of visual representation, although its creation and management of story content is at least tangible. Connections are done with a simple yet dedicated panel for directional linking, but this provides a restricted view to the linkage of a single element. This lack of a complete picture could contribute to why participants rarely reconnected rooms once their initial connection was complete.

5.5 Principle of Branching

Interfaces that use a visual metaphor to represent story structure and connectedness result in the creation of more branches.

In *Articy*, all but one participant created at least one branch with the median being two and the maximum being three for two participants. In contrast, *Inform* only had one participant create a single branch, and in *Quest*, four people created a single branch. We must consider the possibility that differences in branching come from the types of stories that authors were creating. However, given the difference between *Inform* and *Quest*, which both create similar types of stories, this is unlikely to be the only reason. As with the *Principle of Experimentation*, participants using *Articy* were observed to reconfigure branches more than one time, but in the other two programs, a branch was finalized once it was figured out. It is likely that the visual graph and strong first-party support for sequenced dialogue nodes contributed to the number of branches that participants created. Branches in *Inform* are created purely through code. *Inform's* static map mentioned in the *Principle of Structure* was used by participants to assist with identifying and debugging branches, but it is hidden deep in menus, and only updates after successfully compiling the story, and therefore cannot be used at all if an error occurs elsewhere in the compilation process. Branches created with rules are only visible in code and never show up anywhere else. The only way to verify if these work is by testing that part of the story. This lack of visual representation and abstraction means that it is more difficult to create and manage branches in a user-friendly way, especially for non-technical authors. As mentioned in the *Principle of Structure*, a number of participants requested a dynamic graph which could be used for easier visual branch creation. *Quest* is less visual than *Articy* but less abstract than *Inform*. Connections are made in a room's Exit page with labeled buttons representing directions. This is more manageable than *Inform* as there is a tangible object relating to the room based branch, but less so than in *Articy* as only a single set of outputs from a given location is visible at once without context of the surrounding connectivity. Branches resulting from conditional statements are likewise only present in a single script editor view at a time.

5.6 Principle of Contented Authoring

Users that become content with a simple method of authoring are unlikely to explore more advanced features, even if it would enhance their work.

All but one participant using *Articy* only used Dialogues (or Dialogue Fragments), Instructions, and Conditions with simple connectivity. This simple method of authoring satisfied their needs and they did not attempt to explore nor query about the more advanced features within the software. It is likely that the intuition by many upon seeing a flowchart is that there is a clear node-by-node progression, as well as that dialogue, which is centric to this form of storytelling, is readily available and obviously named in the interface. This could result in simple chains of dialogue being the default goto for many users. In both *Quest* and *Inform*, it is less clear what the 'obvious' method of authoring is. There is no initial graph in both software and the way in which dialogue could be implemented in initially unclear. With this lack of clarity, participants had to explore and experiment from the beginning and as a result asked more questions about the advanced features. In *Quest*, three participants went beyond the expected authoring process by adding custom verbs and commands, but most participants didn't notice that they existed. In *Inform*, four participants added more advanced rules, which while higher than the other programs, still remains low. Given that the only way of doing this in *Inform* is through syntax, it is surprising to see. It is possible that without a visual harness, participants felt more willing to try adventurous interactivity, whereas when provided with an obvious method of authoring, they felt content without the need to explore further. Providing a simple and intuitive authoring method to users is a good thing, as it reduces complexity and makes the program more accessible. However, if users become content with the simplified methods of authoring presented, then they may be less likely to explore the more advanced features available. Having a primary method of authoring that enforces a given style could inadvertently limit creativity in that people unknowingly become enclosed within this style.

5.7 Principle of Object Creation

The more of a clear purpose and tangible presence an object/entity has in an interface, the more often they will be created and used.

In all three tools, objects can be referred to purely in text (such as referencing something during speech), but this does not create an actual object instance, and therefore does not allow properties nor interactions to occur with it. In *Inform*, only four people created at least one object, with one creating three (although they remained unused beyond creation), and the remainder not exploring objects at all. Creation is done purely through syntax, the ease of which was praised by two participants. As with most elements in *Inform*, the source code and Index (which was unused by all participants) are the only real reference for tracking objects, and there is no tangible element you can select for editing the object due to the language being compiled. It is therefore not surprising that the object creation rate in *Inform* is low due to the concept of an object instance being quite abstract.

In *Quest*, objects are first class as most elements are a specialization of a generic object type, there are many different ways to create them, and when created they have tangible interface elements that can be selected and interacted with. *Quest* consequently had the highest number of objects created by quite a margin, with every participant creating at least one object with the median being two.

Nobody using *Articy* created tangible objects. Creating objects in *Articy* is done by using an Entities panel to instantiate a specific subtype of Entity. Objects in *Articy* were referenced to in text only without any actual tangible objects being created. Interaction with objects was done instead through the creation and use of variables rather than creating objects explicitly. This may be due to the dialogue-centric approach as described in the *Principle of Contented Authoring*, in that those that are content with an approach may not attempt to add further interactivity, and therefore participants may not have had use for tangible objects as they found, in this context, variables describing states to be suitable replacements.

5.8 Discussion

The above principles are derived from empirical evidence. They are intended to serve as guidance for development and design of future authoring tools so that we can make more informed decisions about our interfaces and UX choices. They also serve as an initial vocabulary to aid further discussion within this space of the UX issues facing authoring tool design. We refer to this listing as ‘principles’ rather than heuristics as we believe that they represent observations of user behavior in response to UX design choices rather than a specific heuristic-based best-practices listing or evaluation framework.

In comparison to established usability and interaction design heuristics such as the works of Nielsen [43], our principles differ in that they aid us in understanding the impact of design choices on domain-specific processes rather than providing general rules of thumb applicable to a broad range of interaction design scenarios. Nielsen’s heuristics, although derived from computing and widely applied to graphical user interfaces, are in fact broad rules of thumb for what makes interaction design in general, regardless of the medium, easy or difficult to use. Due to the nonspecific descriptions of these heuristics, the substance of the definitions must be interpreted for each specific application. For example, if we consider Nielsen’s *Visibility of System Status* heuristic, which encourages designs to inform users of system status through appropriate feedback, the concept of visibility must be interpreted per application. Visibility in mobile software could refer to ensuring the user knows that content is being loaded after a button was tapped, whereas for a voice assistant, visibility may refer to informing the user of state changes through auditory feedback such as sound effects or spoken words. Nielsen also provides an example of this heuristic applied to physical maps with the inclusion of a “You Are Here” icon, where visibility of status refers to the reader's location within the physical environment represented on the map. Consequently, applying the heuristics in any given scenario produces benefits at the lowest common denominator across all interaction designs, which is the lowering of cognitive friction and the increase in usability. However, following these rules of thumb do not then describe the intricacies of a domain-specific process and the response that such a process may generate, as that is not, by definition, shared between all forms of interaction design. To further illustrate the differences between our principles and established heuristics, let us consider a hypothetical design for a new authoring tool. If, when planning the design, developers consider our identified principles, then they will gain insight into the potential response of authors given a certain kind of feature. For example, by considering the *Principle of Metaphor Testing*, developers can work to visually mimic story structure in their design, with the impact being a reduction of reliance upon a dedicated testing harness due to the enhanced visual feedback as described earlier. This takes UX guidance beyond the general principles presented elsewhere towards domain-specific guidance for authoring tools backed by evidence. Existing heuristics for best practices should continue to be used in conjunction with our principles, in that the individual designs that developers come up with remain candidates for refinement of usability. On the other hand, if developers follow established heuristics in isolation, the designs that they come up with may follow best practices, but developers will remain without information as to how authors may respond to such design choices, which is what our principles aim to provide. Further demonstrations for the need of domain-specific UX principles and heuristics can be found elsewhere in the creative and cultural fields connected with technology in Desurvire's work on HEP [10], GAP [12], PLAY [13], and VR PLAY [11], all of which expanded upon general usability heuristics (including those by Nielsen) and other established guidelines to create heuristics lists specific to game playability (HEP), game approachability (GAP), game usability (PLAY), and best practices for virtual reality experiences (VR PLAY). Similarly, Korhonen developed his own heuristics and principles for effective UX in interactive mobile entertainment [34], in which he both subsumed and expanded upon the more generic principles of Nielsen. In both cases, we can see that general UX principles such as Nielsen's *Visibility of System Status* are refined with domain-specific principles such as "Indicators Are Visible" (Korhonen) or "Status Score Indicators Are Seamless, Obvious, Available and Do Not Interfere With Game Play" (Desurvire). Our own principles continue in this tradition providing principles for authoring tools. We do not merely encourage understandable feedback as per Nielsen's *Visibility of System Status*, but specify the importance of visual story metaphors and narrative structure, amongst others. The existence of such works, including our own contributed principles, do not replace or otherwise diminish the value of general usability heuristics, but do show the need for more granularity when dealing with particular domains, as general guidelines are inherently limited in scope due to their abstract nature. In developing domain-specific methods and principles, we are following Greenberg's [22] call to not dogmatically follow established general UX understanding, but to instead formulate methods and understanding specific to the applications in question.

The principles should also be considered in the context of using an adaptation of LRRH as a story template. The critical element in this adaptation was the presence of tropes that are typically found in existing works within the same domain rather than the semantics of the content itself. As mentioned in §2.3, this process ensured that participants would interact with the authoring tools in a representative environment for authoring of interactive narrative, that a reasonable effort could be made to complete the task within time while still allowing for substantial interactions with the tools, and to mitigate potential time sinks that may occur by asking participants to rapidly create story material without reference. It is possible that the particular structure that emerged from the LRRH adaptation had an impact upon the way that participants interacted with the authoring tools. The story was there mostly to provide a basis for interacting with the tools, and there is no direct evidence that the story style or premise directly impacted the principles we include, but naturally it remains possible that it influenced outcomes as a methodological context, as with any UX study. However, exploring the specific impact of differing story structures upon authoring tool usage is beyond the scope of this study which seeks to establish initial principles for this domain. Similarly, it is possible that removing a different section of the story for participants to complete could have influenced their interactions with the authoring tools. This is why, as described in §3, we considered the complexity of each section, the sufficiency of interactions with the authoring tool that this could result in, and the time it would take to reasonably attempt to complete the section given the time restrictions.

These principles, although established within an interactive storytelling context, are applicable to the wider cultural heritage authoring tool development community. As discussed in §3, authoring tools for cultural heritage are developed with creatives in mind that typically do not have technical backgrounds, such as museum curators or archaeological guides, and therefore share a common interest of target users with interactive storytelling. Additionally, many interactive experiences created in other cultural heritage domains share properties with interactive storytelling. For example, triggering events based upon user interaction [3] or including branching pathways in the experience [23, 38, 49]. In some cases, cultural heritage works from outside of interactive storytelling build directly upon its foundations [23]. In previous work [19], we enumerated various UI paradigms within IDN authoring tools which can also be found within other branches of cultural heritage authoring tool design. For example, the *CHESS* authoring tool used by Roussou et al. [49] primarily uses graph-based editing and multifaceted form fields for data entry, much alike *Articy* in our study. Similarly, interfaces that are primarily multifaceted are commonplace such as in *Cicero Designer* [16] and most authoring tools discussed in a review by Fidas et al. [15], which are similar to *Quest* in our study. We have, however, been unable to find a clear parallel to text-based interfaces as seen in *Inform* in our study, perhaps due to the coding paradigm followed by domain-specific languages potentially being less accessible to the target users within the cultural heritage domain. This is discussed in more detail in §2.1.4. It is also worth considering that the authoring tools within our broad selection are either desktop applications or web applications. Consequently, the user interface and interaction paradigms that authors were exposed to during the study likely differ from those found in tabletop authoring tools and in some mobile authoring tools that make use of features exclusive to mobile devices. Despite this potential limitation, the principles are still applicable in circumstances where visual presentation and interactions overlap. For example, while the details of interacting with a node graph for content creation on a mobile authoring tool may differ from those found on an equivalent desktop application, the fundamental concept of visually representing the narrative structure as a node graph remains complementary. The details of how user interface and interaction design between disciplines impact an authoring experience is beyond the scope of this study, but should be considered when applying to domains that we did not directly touch upon.

6 CONCLUSION

This article set out to identify the impact of different user interface paradigms on the UX of IDN authoring tools. Evaluating authoring tool UX presents significant challenges, as unlike with many short task-focused web applications (which much of contemporary UX practice focuses upon), typical usage is not measured in short interactions that can be completed in minutes but instead in substantial creative endeavours that take months or even years. Consequently, the challenges in evaluating these systems has led to an absence of research exploring the impact of the UX of authoring tool technology despite its importance, particularly in domains with a strong connection to interactive digital narrative such as computational cultural heritage. This gap has served as the motivation for this study. We tackled our research question by conducting a user study using a novel bespoke evaluation methodology for IDN authoring tools which addresses the difficulties facing authoring tool evaluation. While our method addresses these challenges, it does introduce some compromises and assumptions, and it is important to recognise our conclusions in context of the constraints of our method. Our method involves authoring tasks that are substantially shorter than full authorship and often did not permit participants to complete fully-functional stories, and as such may be considered a somewhat stressed UX. Furthermore, it is possible that the genre and style of our template story may have influenced the UX. Our method does take steps to mitigate these issues, however, by using informed and representative story features, and presenting the participants with a template to work from such that within the bounds of our study they interact with a range of authoring tool features in a reasonable writing context. Consequently, while it is important that our conclusions are considered within the constraints of the methodological design, we believe that the overall methodology is suitable and that valuable conclusions can be produced from studies of this kind. In forming bespoke methodologies for this challenging domain, we are following Greenberg's call [22] for evaluation methods to be designed for the constraints and context of the application being studied, not just following previous methods established for tools in a very different context.

The study was oriented around interviewing authors and observing their behaviors within interactive narrative authoring tools that exhibit varying interface paradigms in order to identify any trends that demonstrate the impact that these differences make upon authoring workflow. Included was our approach to choosing a subset of representative authoring tools from the wider selection. This method used hierarchical clustering of various identified features in order to determine groupings, and then individuals from within those groups were chosen. This was done to ensure that the selected subset of authoring tools were a better representation of the global feature space than choosing at random. We took a similar approach for the selection of tasks, basing them on a frequency analysis of existing storytelling features found within a wide variety of video games. These tasks were the foundation for the structure of the LRRH script, as well as for the tasks given to participants as guidance. Following was an in-depth description of the employed methodology, our procedures for gathering data from the experiment, and discussion of the techniques behind our data analysis, particularly with respect to how we looked to find supporting information for our definition of authoring workflow.

Our seven principles (Metaphor Testing, Fast Track Testing, Structure, Experimentation, Branching, Contented Authoring, Object Creation) highlight relationships between design of user interfaces in IDN authoring tools and their potential impact upon the workflow of authors. These principles are intended to serve as guidance for future authoring tools and a vocabulary for this field to further discuss the UX issues facing authorship technology. This enables us to identify how different design decisions for authoring tools hinder or aid the authors, and provides us with an understanding of how the decisions may impact the authors. This can help us as developers to make more intelligent decisions about our interfaces and UX choices, which ultimately serves to create better authoring tools that can reach and better support a wider audience.

The findings that we have presented come from a representative yet constrained workflow for interactive storytelling authorship. While we conclude our work here with a set of principles for authoring tools based on different interface paradigms and account for representative tools and story features, it is possible that a number of other contributing variables, such as story genre, may have impact. It is always possible that methodological contexts may influence findings in UX studies, but in the context of our work, which focuses on the use of authoring tools, we do not see any evidence that this context has had a direct impact on our conclusions. We feel the principles and observations remain valuable, but should but understood in the context of the methods that lead to their discovery and with the acknowledgement that we do not here explore the impact of specific story styles or genres. While a study of UX principles by genre, application, delivery platform, or other story types is beyond the scope of this work, we do feel that this represents an exciting and important priority for future work in this space either by seeking to account for this in new methods or creating bespoke studies that validate or change UX principles by individual story styles.

REFERENCES

[1] Ernest Adams. 2014. *Fundamentals of Game Design* (3rd ed.). New Riders Publishing.

[2] Roberto Andreoli, Angela Corolla, Armando Faggiano, Delfina Malandrino, Donato Pirozzi, Mirta Ranaldi, Gianluca Santangelo, and Vittorio Scarano. 2018. A framework to design, develop, and evaluate immersive and collaborative serious games in cultural heritage. *Journal on Computing and Cultural Heritage (JOCCH)* 11, 1 (2018), 4.

[3] C. Ardito, M. F. Costabile, G. Desolda, M. Matera, A. Piccinno, and M. Picozzi. 2012-10-14. Composition of situational interactive spaces by end users: a case for cultural heritage. In *Proceedings of the 7th Nordic Conference on Human-Computer Interaction: Making Sense Through Design* (New York, NY, USA) *(NordiCHI '12)*. Association for Computing Machinery, 79–88. https://doi.org/10.1145/2399016.2399029

[4] Mafkereseb Kassahun Bekele, Roberto Pierdicca, Emanuele Frontoni, Eva Savina Malinverni, and James Gain. 2018. A survey of augmented, virtual, and mixed reality for cultural heritage. *Journal on Computing and Cultural Heritage (JOCCH)* 11, 2 (2018), 7.

[5] Mark Bernstein. 2016. Storyspace 3. In *Proceedings of the 27th ACM Conference on Hypertext and Social Media* (New York, NY, USA) *(HT '16)*. ACM, 201–206. https://doi.org/10.1145/2914586.2914624

[6] Mark Blythe, Josephine Reid, Peter Wright, and Erik Geelhoed. 2006. Interdisciplinary criticism: analysing the experience of riot! a location-sensitive digital narrative. *Behaviour & Information Technology* 25, 2 (2006), 127–139.

[7] Wayne C Booth. 1961. *The Rhetoric of Fiction*. University of Chicago Press.

[8] Fred Charles, David Pizzi, Marc Cavazza, Thurid Vogt, and Elisabeth André. 2009. EmoEmma: Emotional Speech Input for Interactive Storytelling. In *Proceedings of The 8th International Conference on Autonomous Agents and Multiagent Systems - Volume 2* (Budapest, Hungary) *(AAMAS '09)*. International Foundation for Autonomous Agents and Multiagent Systems, Richland, SC, 1381–1382. http://dl.acm.org/citation.cfm?id=1558109.1558305

[9] Alan Cooper. 2004. *The Inmates Are Running the Asylum: Why High Tech Products Drive Us Crazy and How to Restore the Sanity (2nd Edition)*. Pearson Higher Education.

[10] Heather Desurvire, Martin Caplan, and Jozsef A. Toth. 2004-04-24. Using Heuristics to Evaluate the Playability of Games. In *CHI '04 Extended Abstracts on Human Factors in Computing Systems* (New York, NY, USA) *(CHI EA '04)*. Association for Computing Machinery, 1509–1512. https://doi.org/10.1145/985921.986102

[11] Heather Desurvire and Max Kreminski. 2018. Are Game Design and User Research Guidelines Specific to Virtual Reality Effective in Creating a More Optimal Player Experience? Yes, VR PLAY. In *Design, User Experience, and Usability: Theory and Practice* (Cham) *(Lecture Notes in Computer Science)*, Aaron Marcus and Wentao Wang (Eds.). Springer International Publishing, 40–59. https://doi.org/10.1007/978-3-319-91797-9_4

[12] Heather Desurvire and Charlotte Wiberg. 2008. Master of the Game: Assessing Approachability in Future Game Design. In *CHI '08 Extended Abstracts on Human Factors in Computing Systems* (New York, NY, USA) *(CHI EA '08)*. Association for Computing Machinery, 3177–3182. https://doi.org/10.1145/1358628.1358827

[13] Heather Desurvire and Charlotte Wiberg. 2009. Game usability heuristics (PLAY) for evaluating and designing better games: The next iteration. In *International conference on online communities and social computing*. Springer, 557–566.

[14] Laura Ermi and Frans Mäyrä. 2005. Fundamental components of the gameplay experience: Analysing immersion. *Worlds in play: International perspectives on digital games research* 37, 2 (2005), 37–53.

[15] Christos Fidas, Christos Sintoris, Nikoleta Yiannoutsou, and Nikolaos Avouris. 2015. A Survey on Tools for End User Authoring of Mobile Applications for Cultural Heritage. In *2015 6th International Conference on Information, Intelligence, Systems and Applications (IISA)*. 1–5. https://doi.org/10.1109/IISA.2015.7388029

[16] Giuseppe Ghiani, Fabio Paternò, and Lucio Davide Spano. 2009. Cicero Designer: An Environment for End-User Development of Multi-Device Museum Guides. In *End-User Development* (Berlin, Heidelberg) *(Lecture Notes in Computer Science)*, Volkmar Pipek, Mary Beth Rosson, Boris de Ruyter, and Volker Wulf (Eds.). Springer, 265–274. https://doi.org/10.1007/978-3-642-00427-8_15

[17] Stefan Göbel, Luca Salvatore, and Robert Konrad. 2008. StoryTec: A digital storytelling platform for the authoring and experiencing of interactive and non-linear stories. In *2008 International Conference on Automated Solutions for Cross Media Content and Multi-Channel Distribution*. Ieee, 103–110.

[18] Elizabeth Goodman, Mike Kuniavsky, and Andrea Moed. 2012. *Observing the user experience: A practitioner's guide to user research*. Elsevier.

[19] Daniel Green, Charlie Hargood, and Fred Charles. 2018. Contemporary Issues in Interactive Storytelling Authoring Systems. In *Interactive Storytelling* (Cham) *(Lecture Notes in Computer Science)*. Springer International Publishing, 501–513. https://doi.org/10.1007/978-3-030-04028-4_59

[20] Daniel Green, Charlie Hargood, and Fred Charles. 2018. Define "Authoring Tool": A Survey of Interactive Narrative Authoring Tools. In *Authoring for Interactive Storytelling Workshop, ICIDS*.

[21] Daniel Green, Charlie Hargood, Fred Charles, and Alexander Jones. 2018. Novella: A Proposition for Game-Based Storytelling. In *Narrative and Hypertext 2018*. ACM, Baltimore, Maryland.

[22] Saul Greenberg and Bill Buxton. 2008. Usability Evaluation Considered Harmful (Some of the Time). In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Florence, Italy) *(CHI '08)*. Association for Computing Machinery, 111–120. https://doi.org/10.1145/1357054.1357074

[23] Frank Allan Hansen, Karen Johanne Kortbek, and Kaj Grønbæk. 2012-03-01. Mobile Urban Drama: interactive storytelling in real world environments. 18, 1 (2012-03-01), 63–89. https://doi.org/10.1080/13614568.2012.617842

[24] Charlie Hargood, Mark J. Weal, and David E. Millard. 2018. The Storyplaces Platform: Building a Web-Based Locative Hypertext System. In *Proceedings of the 29th ACM Conference on Hypertext and Social Media (HT '18)*. ACM, New York, NY, USA.

[25] Christopher L. Holden, David J. Gagnon, Breanne K. Litts, and Garrett Smith. 2014. ARIS: An Open-Source Platform for Widespread Mobile Augmented Reality Experimentation. In *Technology Platform Innovations and Forthcoming Trends in Ubiquitous Learning*. IGI Global, 19–34. https://doi.org/10.4018/978-1-4666-4542-4.ch002

[26] Henry Jenkins. 2004. Game Design as Narrative Architecture. *First Person: New Media as Story, Performance, and Game* 44 (2004).

[27] Catherine Emma Kate Jones, Stathis Theodosis, and Ioanna Lykourentzou. 2019. The Enthusiast, the Interested, the Sceptic, and the Cynic: Understanding User Experience and Perceived Value in Location-Based Cultural Heritage Games Through Qualitative and Sentiment Analysis. *Journal on Computing and Cultural Heritage (JOCCH)* 12, 1 (2019), 6.

[28] Akrivi Katifori, Manos Karvounis, Vassilis Kourtis, Sara Perry, Maria Roussou, and Yannis Ioanidis. 2018. Applying Interactive Storytelling in Cultural Heritage: Opportunities, Challenges and Lessons Learned. In *International Conference on Interactive Digital Storytelling*. Springer, 603–612.

[29] Akrivi Katifori, Maria Roussou, Sara Perry, George Drettakis, Sebastian Vizcay, and Julien Philip. [n.d.]. The EMOTIVE Project-Emotive Virtual Cultural Experiences through Personalized Storytelling. In *CIRA@EuroMed* (2018). 11–20.

[30] Hartmut Koenitz. 2011. Extensible Tools for Practical Experiments in IDN: The Advanced Stories Authoring and Presentation System. In *Proceedings of the 4th International Conference on Interactive Digital Storytelling (ICIDS'11)*. Springer, Berlin, Heidelberg, 79–84.

[31] Markos Konstantakis and George Caridakis. 2020. Adding Culture to UX: UX Research Methodologies and Applications in Cultural Heritage. *J. Comput. Cult. Herit.* 13, 1, Article 4 (Feb. 2020), 17 pages.

[32] Seungbum Koo, Jinyoung Kim, Changhyuk Kim, Jeeyeop Kim, and Hee Sung Cha. 2019. Development of an Augmented Reality Tour Guide for a Cultural Heritage Site. *Journal on Computing and Cultural Heritage (JOCCH)* 12, 4 (2019), 1–24.

[33] Seungbum Koo, Jinyoung Kim, Changhyuk Kim, Jeeyeop Kim, and Hee Sung Cha. 2019. Development of an Augmented Reality Tour Guide for a Cultural Heritage Site. *Journal on Computing and Cultural Heritage (JOCCH)* 12, 4 (2019), 24.

[34] Hannu Korhonen. 2016. *Evaluating playability of mobile games with the expert review method*. Ph.D. Dissertation. Tampere University.

[35] Sébastien Lê, Julie Josse, François Husson, et al. 2008. FactoMineR: An R Package for Multivariate Analysis. *Journal of Statistical Software* 25, 1 (2008), 1–18.

[36] Henry Lieberman, Fabio Paternò, Markus Klann, and Volker Wulf. 2006. End-User Development: An Emerging Paradigm. In *End User Development*, Henry Lieberman, Fabio Paternò, and Volker Wulf (Eds.). Springer Netherlands, 1–8. https://doi.org/10.1007/1-4020-5386-X_1

[37] Gunnar Liestøl. 2014. Along the Appian Way. Storytelling and memory across time and space in mobile augmented reality. In *Euro-Mediterranean Conference*. Springer, 248–257.

[38] M. T. Linaza, I. Torre, R. Beusing, A. Tavernise, and M. Etz. 2008-12-02. Authoring Tools for Archaeological Mobile Guides. In *Proceedings of the 9th International conference on Virtual Reality, Archaeology and Cultural Heritage* (Goslar, DEU) *(VAST'08)*. Eurographics Association, 47–54.

[39] Stacey Mason. 2013. On Games and Links: Extending the Vocabulary of Agency and Immersion in Interactive Narratives. In *Interactive Storytelling (Lecture Notes in Computer Science)*. Springer International Publishing, 25–34.

[40] David E Millard and Charlie Hargood. 2017. Tiree Tales: A Co-operative Inquiry into the Poetics of Location-Based Narrative. In *Proceedings of the 28th ACM Conference on Hypertext and Social Media*. ACM, 15–24.

[41] David E Millard, Charlie Hargood, Yvonne Howard, and Heather Packer. 2017. The StoryPlaces Authoring Tool: Pattern Centric Authoring. In *Authoring for Interactive Storytelling Workshop, ICIDS*.

[42] David E Millard, Charlie Hargood, Michael O Jewell, and Mark J Weal. 2013. Canyons, deltas and plains: towards a unified sculptural model of location-based hypertext. In *Proceedings of the 24th ACM Conference on Hypertext and Social Media*. ACM, 109–118.

[43] Jakob Nielsen. 1994. Enhancing the Explanatory Power of Usability Heuristics. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '94)*. ACM, New York, NY, USA, 152–158.

[44] Eva Pietroni, Alfonsina Pagano, and Claudio Rufa. 2013. The Etruscanning project: Gesture-based interaction and user experience in the virtual reconstruction of the Regolini-Galassi tomb. In *2013 Digital Heritage International Congress*, Vol. 2. IEEE, 653–660.

[45] Johanna Pirker, Christian Gütl, Patrick Weiner, Victor Manuel Garcia-Barrios, and Melanie Tomintz. 2014-11. Location-Based Mobile Application Creator Creating Educational Mobile Scavenger Hunts. In *2014 International Conference on Interactive Mobile Communication Technologies and Learning (IMCL2014)*. 160–164. https://doi.org/10.1109/IMCTL.2014.7011124

[46] D. A. Plecher, M. Wandinger, and G. Klinker. 2019. Mixed Reality for Cultural Heritage. In *2019 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*. 1618–1622. https://doi.org/10.1109/VR.2019.8797846

[47] Laia Pujol-Tost. 2019. Did We Just Travel to the Past? Building and Evaluating With Cultural Presence Different Modes of VR-Mediated Experiences in Virtual Archaeology. *Journal on Computing and Cultural Heritage (JOCCH)* 12, 1 (2019), 2.

[48] George E Raptis, Christos Fidas, and Nikolaos Avouris. 2019. Do Game Designers' Decisions Related to Visual Activities Affect Knowledge Acquisition in Cultural Heritage Games? An Evaluation From a Human Cognitive Processing Perspective. *Journal on Computing and Cultural Heritage (JOCCH)* 12, 1 (2019), 4.

[49] Maria Roussou, Laia Pujol, Akrivi Katifori, Angeliki Chrysanthi, Sara Perry, and Maria Vayanou. 2015. The Museum As Digital Storyteller: Collaborative Participatory Creation of Interactive Digital Experiences. (2015). https://eprints.gla.ac.uk/143243/

[50] Katie Salen and Eric Zimmerman. 2004. *Rules of Play: Game Design Fundamentals*. MIT Press.
[51] Guy Schofield, Gareth Beale, Nicole Beale, Martin Fell, Dawn Hadley, Jonathan Hook, Damian Murphy, Julian Richards, and Lewis Thresh. 2018. Viking VR: Designing a Virtual Reality Experience for a Museum. In *Proceedings of the 2018 Designing Interactive Systems Conference* (Hong Kong, China) *(DIS '18)*. ACM, New York, NY, USA, 805–815. https://doi.org/10.1145/3196709.3196714
[52] Elmedin Selmanovic, Selma Rizvic, Carlo Harvey, Dusanka Boskovic, Vedad Hulusic, Malek Chahin, and Sanda Sljivo. 2018. VR Video Storytelling for Intangible Cultural Heritage Preservation. In *Eurographics Workshop on Graphics and Cultural Heritage*. The Eurographics Association. https://doi.org/10.2312/gch.20181341
[53] Yotam Shibolet, Noam Knoller, and Hartmut Koenitz. 2018. A framework for classifying and describing authoring tools for interactive digital narrative. In *International Conference on Interactive Digital Storytelling*. Springer, 523–533.
[54] Christos Sintoris, Nikoleta Yiannoutsou, Alejandro Ortega-Arranz, Rodrigo López-Romero, Menita Masoura, Nikolaos Avouris, and Yannis Dimitriadis. 2014-11. TaggingCreaditor: A tool to create and share content for location-based games for learning. In *2014 International Conference on Interactive Mobile Communication Technologies and Learning (IMCL2014)*. 280–284. https://doi.org/10.1109/IMCTL.2014.7011148
[55] Ulrike Spierling, Peter Winzer, and Erik Massarczyk. 2017. Experiencing the presence of historical stories with location-based augmented reality. In *International Conference on Interactive Digital Storytelling*. Springer, 49–62.
[56] Ektor Vrettakis, Vassilis Kourtis, Akrivi Katifori, Manos Karvounis, Christos Lougiakis, and Yannis Ioannidis. 2019. Narralive–Creating and experiencing mobile digital storytelling in cultural heritage. *Digital Applications in Archaeology and Cultural Heritage* 15 (2019).

A APPENDIX A

A.1 Story Features

Choices (`None/Diegetic/Extra-Diegetic`)
Are choices diegetic (without suspending story flow) or extra-diegetic (suspension of flow during choice)?

Resolve (`Linear/Branching/Perpetual`)
How does the story end? Is it linear, branching, or does it not end?

Narrator Type (`None/Dramatized/Undramatized`)
How dramatized the narrators are, based on Booth [7].

Narrator Significance (`None/Substantial/Insubstantial`)
Whether the narrator plays a key role in the story or not.

Plot Progression (`Task/Interaction`)
The main method to progress story. Sets of tasks may gate progress, holding the player back until completed. Otherwise, simple interactions may progress the story.

Discoverable Narrative INEA (`Yes/No`)
Does the game feature Intangible/Narrative/Explicit/Active elements of Discoverable Narrative?

Discoverable Narrative TNEA (`Yes/No`)
Does the game feature Tangible/Narrative/Explicit/Active elements of Discoverable Narrative?

Cutscene (`Yes/No`)
Does the game have cutscenes?

Interruptible Cutscene (`Yes/No`)
Does the game have cutscenes that can be interrupted or influenced?

Movie (`Yes/No`)
Does the game feature cinematic sequences?

In-Game Event (`Yes/No`)
Does the game break regular behavior of non-playable characters or objects for the purpose of narrative.

Player Traits Alter Narrative (`Yes`/`No`)
Do player traits alter the narrative, such as morality points affecting which conversation options are available?

Mechanics as Metaphor (`Yes`/`No`)
Does the game use mechanics as a metaphor?

Environmental Storytelling (`Yes`/`No`)
Does the game feature environmental storytelling?

A.2 Demographics Questionnaire

What is your profession or field of study?
An enumeration of both Student and Professional (Creative, Technical, Other).

What was your previous knowledge of the tool before this experiment?
Either `None`, `Little`, `Moderate`, or `Plenty`.

What is your previous knowledge of any other similar tools before this experiment?
Either `None`, `Little`, `Moderate`, or `Plenty`.

What was your previous knowledge of interactive digital narrative before this experiment?
Either `None`, `Little`, `Moderate`, or `Plenty`.

Was the short training video adequate as a brief introduction to the software?
Either `Yes` or `No`.

A.3 Interview Questions

Tell me about how you felt the features in the tool were a hindrance to your goal of implementing your vision. What was it that frustrated you?
What elements of the tool caused a negative experience for this participant?

Tell me about how you felt the features in the tool aided your goal of implementing your vision. What was it that satisfied you?
What elements of the tool caused a positive experience for this participant?

If you could add any features to make implementing your vision easier, what would they be and why?
Discover what elements the participant would like to add that were not present.

If you could alter any existing features to make implementing your vision easier, what would they be and why?
Discover what already existing elements in the tool the participants would change.

How would you have completed or improved your implemented solution given more time?
Determine how the participant would have continued refining or completing their solution given more time.

A.4 Analysis Actions

Action Description (A) - Text ‘spoken’ by a narrator describing the state of the world with respect to a character’s action or state.

Authored Branch Creation (B) - A branch is purposely created by the author to either provide explicit choice between distinct pathways, or to be implicitly redirected along two or more pathways. Branches must be of narrative significance. For example, having an object present in a location that is optional to examine does not constitute as a branch. In the case of adventure games, branching refers to rooms connected to multiple rooms with purpose, and conditional pathways based on state checks.

Character Creation (C) - Adding a new character to the story. Characters must be distinct and not only mentioned in text. This includes animals.

Documentation (D) - The author accesses documentation. This includes the official documentation as well as referencing back to the training video or searching the internet.

Guidance (G) - The author is guided by the experimenter to avoid deadlock without explicitly asking for help. Guidance of comes from a Query but differs in that Query has no following assistance on the topic.

Help (H) - The author explicitly asks the experimenter for assistance.

Advanced Interactivity (I) - Adding behavior other than basic defaults to objects and the world. This includes adding custom rules to a world, custom attributes and commands, as well as more complex linking methods.

Location Creation (L) - A location or room is created or initially specified. This includes renaming and repurposing of temporary rooms, as well as using new 'Flow' nodes to segment stories.

Object Creation (O) - Adding a new entity to the story. Entities must be distinct and not only mentioned.

Query (Q) - The author asks for clarification on something in the program, but does not ask for assistance.

Character Speech (S) - A character explicitly speaks. This can be done in text as long as speaker is known.

Testing (T) - The author tests the story from within the program for previewing or debugging purposes.

Variable/Script Creation (V) - A variable or script is initially created for fine control over the story state.

World Building (W) - Present text to the player that describes and conveys the state of the world.

A.5 Axial Coding Categories

Bold text is the purpose, italic is the observation, and the sentence following is the classification criteria.

Which tool had the most UI elements that caused confusion?
Unclear UI elements cause confusion.
Mention of an interface element or feature that is unclear, resulting in confusion and/or difficulty.

Do participants like graph systems?
The graph system is good/helpful/etc.
Positive mention of a graph's functionality and helpfulness.

Did people want intelligent assistance for writing code?
Inclusion of an intelligent assistant when writing code.
Mention of the desire for assistance when writing code, such as autocomplete or linting.

Are parts of the interface too small?
Elements of the interface are too small.
Comments on the size of elements in the interface being less than adequate.

Do people like the ability to return to a specific state during testing?
Assistance to return to a specific state during testing.
Discussion on how a state-returning feature (play for here, Skein, Walkthrough) is helpful.

Which tool did people comment on being easier to use?
Using the tool is easy having gotten used to it.
Mention of the ease of use as familiarity increases.

Is wording used in labels a cause of confusion?
Unclear terminology caused confusion.
Mention of terminology used in labels (and such) that resulted in confusion or difficulty of understanding.

Are errors unclear to authors?
Error messages are difficult to understand.
Complaints about the clarity or helpfulness of errors (including compile, runtime, and system).

How many people thought variables were useful?
Variables are useful.
Positive mention on the functionality of variables.

Did people want support for more advanced dialogue systems?
Support for more advanced dialogue functionality than built-in.
Mention of the desire for a dialogue system that extends the capabilities of the tool's existing functionality.

Was there a difficulty in use of the built-in syntax?
Difficulty in use of built-in syntax.
Complaints or concerns about the clarity and form of built-in syntax for scripting.

Did users want a graph system when it wasn't present?
Desire of a graph system when it wasn't present.
Mention of wanting a graph system where it did not exist, or if it did, it was not satisfactory.

Do users find it easy to populate the world?
Populating the world with objects is easy.
Mention of ease of populating the game world.

In which program was it easier to create world structure?
Creation of the world structure is simple.
Praise for the ease of creating world layouts.

Are hierarchies considered useful?
Hierarchies are useful.
Positive comments on the functionality of a hierarchy or outliner.

Do users think that variables should be made more accessible?
Variables should be easier to handle.
Mention of desire to simplify variables.

Which tool was easier to create branches in?
Creation of branches is easy.
Mentions of ease of creating branches regardless of the content.

Was there any praise for the built-in syntax being intuitive?
Intuitive syntax.
Positive comments on the form of the built-in syntax for scripting.

Are participants overwhelmed by certain interfaces?
The overall interface is overwhelming.
Complaints about the complexity of the interface and its lack of understandability as a whole.

Is it easy to edit content?
Editing content is easy.
Praise for ease of editing content of elements such as object descriptions.

Were structural layout shortcuts useful?
Shortcuts were useful to create layouts.
Positive mention of shortcuts for structure creation.

Did people want support for game-centric features that weren't present?
Support for more advanced game-centric features over what's built in.
Mention of the desire for absent functionality that relates to game narrative, such as support for quest systems.

Are context menus favored?
Good use of context menus.
Positive comments on the usefulness of context menus.

Was the existing intelligent assistance for syntax any good?
Intelligent syntax assistance is good.
Positive mention of existing intelligent assistance.