The 13th International Symposium on Virtual Reality, Archaeology and Cultural Heritage VAST (2012)
D. Arnold, J. Kaminski, F. Niccolucci, and A. Stork (Editors)

Authoring animated interactive 3D Museum Exhibits using a Digital Repository

R. Zmugg[1], W. Thaller[1], M. Hecher[2], T. Schiffer[1], S. Havemann[1], D.W. Fellner [1,3]

[1]Institut of ComputerGraphics & KnowledgeVisualization (CGV), TU Graz, Austria
[2]Fraunhofer Austria Research GmbH, Austria
[3]TU Darmstadt & Fraunhofer IGD, Germany

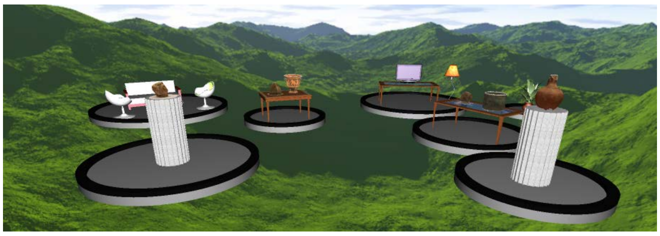

Figure 1: *Interactive three-dimensional animations can convey a meaning and direct user attention. In contrast to single-object presentations they have a plot. Artifacts can be shown in a context, in different configurations, details can be emphasized. Our paper describes an efficient authoring approach that allows curators to actively control and shape the resulting 3D animations.*

Abstract

We present the prototype of a software system to streamline the serial production of simple interactive 3D animations for the display in museum exhibitions. We propose dividing the authoring process in two phases, a designer phase and a curator phase. The designer creates a set of configurable 3D scene templates that fit with the look of the physical exhibition while the curator inserts 3D models and configures the scene templates; the finished scenes are uploaded to 3D kiosks in the museum. Distinguishing features of our system are the tight integration with an asset repository and the simplified scene graph authoring. We demonstrate the usefulness with a few examples.

Categories and Subject Descriptors (according to ACM CCS): I.3.8 [Computer Graphics]: Applications—I.3.4 [Computer Graphics]: Graphics editors—I.3.m [Computer Graphics]: Installations—

Keywords: Cultural heritage, digital exhibitions, virtual artifacts, 3D animation, authoring systems, scene graph

1. Introduction

Visualizations of Cultural Heritage (CH) content with interactive animations, showing dynamically moving objects, triggered and controlled by user interaction, can be a great asset in museum exhibitions. While single-object viewers allow the detailed inspection of a single high-quality artefact, they typically engage the viewer only with embedded information points that bring up additional text and images. Single object viewers show a static object that has no inherent story, no plot. Interactive animations are perceived as more engaging than single object exploration since they can be used in many different ways, for example

- to show an object in its context (excavation site),
- to show how objects were utilized and employed,
- to show related objects physically residing elsewhere,
- to show 3D comparisons with other similar objects,
- to explain intricate assemblies with exploding views,
- to show transitions between different hypotheses, etc.

DOI: 10.2312/VAST/VAST12/073-080

Figure 2: *Presentation of three digital artifacts using the Unity3D game engine. Each object can be inspected in orbit mode. When another object is chosen, a smooth animation ensures cognitive coherence in the transition.*

Figure 3: *The early Neolithic well in Altscherbitz, Saxony, with wood dating from 5102 b.c., i.e., 7100 years old. Because of its good state and importance, it was transferred* en bloc *and excavated in the lab of the Archeological Office.*

Creating compelling animations is tedious and costly even when not counting the cost of 3D acquisition. Using high-end tools like Maya or 3DStudioMax to prepare the content, the auxiliary scene backdrop and the texturing, and using game engines like Virtools or Unity3D (see Fig. 2) for presentation will certainly remain the method of choice for creating high-end visualizations for the foreseeable future. But we observe a massive increase in 3D digitization campaigns that eventually produce large quantities of 3D museum artifacts with greatly improved quality. With literally hundreds of thousands of high-quality artefacts available, the scalability of content production becomes an issue. It is therefore time for research on new ways of producing compelling interactive content in a more cost and labour efficient fashion.

If using 3D really becomes a standard in museums, and physical museum exhibits and showcases are explained using displays rather than paper posters and notes, then these displays can show explanations also in 3D. Since many things are easier to explain using small 3D animations, this opens up a niche for a new type of authoring tools: It must take minutes, not hours, to create these small animations, otherwise their mass production is not feasible.

1.1. Contribution

With the prototype system presented in this paper we propose three main ingredients to streamline the production of animated CH visualizations. First, we follow in two ways a strategy like SketchUp for reducing the tool complexity, (i) by providing a small but sufficient set of *scene graph* manipulation operations represented by graphical widgets (Fig. 8), and (ii) with an integrated push-pull oriented shape modeling engine (Fig. 10). The second main source of efficiency is the direct integration with a digital asset repository, in this case the Repository Infrastructure (RI) of 3D-COFORM (see Section 3.2). And third, we propose dividing the authoring process in two parts with

- a **designer mode** where design-oriented staff creates a set of good-looking configurable *scene templates* containing the scene backdrop, animations, and placeholders instead of 3D assets (*drop targets*, see Section 4), and
- a **curator mode** where one of the available scene templates is chosen and filled via drag-and-drop with high-quality 3D assets from the RI (see Section 3).

1.2. Benefit

Animated 3D visualizations are a powerful means to attract and direct the attention of visitors; they convey knowledge and explanations quickly and in a meaningful way. A whole range of possibilities exist, from those mentioned in the beginning up to storytelling with animated humans.

We realized the usefulness of 'small animations' (as opposed to expensive high-end visualizations) in discussions with archeologists concerned with the conservation of prehistoric wells. The finds in the Altscherbitz well (Fig. 3) tell much about everyday things and their use in Germany in different periods starting from 5100 b.c. The position of the findings is faithfully (3D-)documented since strata from different periods are stacked upon each other. This is difficult to explain by drawings, so the archeologists thought about series of simple exploding views, in fact just scene graph animations, to explain the situation of the findings. Asked for advice, we proposed the usual set of standard all-purpose tools. But the archeologists did not want to learn Maya; instead they asked: *"Isn't there 3D software as simple as SketchUp for creating animations?"* - Altscherbitz is not yet among our results (Sec. 5) but we mention it as our initial motivating example, and to illustrate the benefit.

2. Related Work

Stunning 3D exhibits with high visibility like the interactive exploration of an Egyptian mummy at the British museum

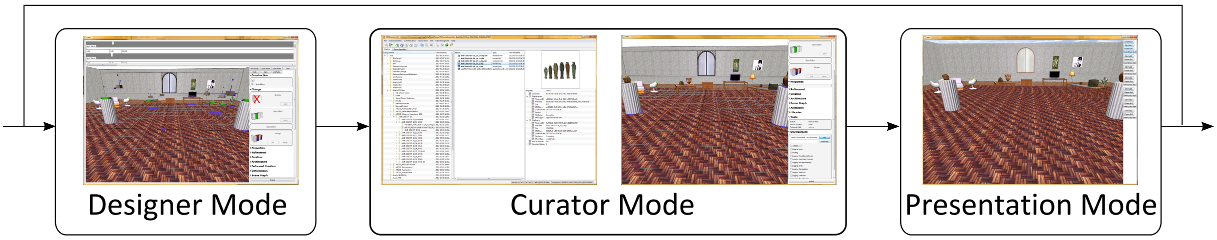

Figure 4: *The production of animated scenes proceeds from designer mode to curator mode, finally to presentation mode.*

[BSly], and many others, have created a demand for 3D visualizations in museums. But for museums to show 3D objects on a regular basis the cost efficiency must be improved without sacrificing quality. Assuming that a large pool of high-quality 3D artefacts exists, nearly a serial production of 3D exhibits is possible with a single-object viewing approach: Skinnable viewers like VirtualInspector [CPCS08, CPCS06] can be configured using good-looking HTML pages and a bit of script code. It was used for many complementary exhibits, e.g., to let visitors virtually explore on a 3D kiosk the chisel marks of the 5 m tall David statue in Florence. The evolving *Colonia 3D* project takes a rather different approach to obtaining efficiency by custom-tailoring the presentation software to their comprehensive collection of thousands of 3D artifacts in the context of a complete historical city reconstruction. Their browser even supports the comparison with panoramas of contemporary Cologne [TSP*12]. They propose three user interaction modes: findings mode, reconstruction mode, and comparison mode.

Much recent work on virtual museums and 3D exhibitions focuses on using game engines because of their superior visual quality compared to the formerly used X3D viewers. To cite two representative approaches, an adaptation of the Torque3D engine with scripted interaction and its level editor adapted to exhibition layout was presented by [MSLV08]; and the VEX-CMS from [CIR*10] based on OGRE features a custom application for exhibition design also using game concepts for object placement, guided tours, and the VOI (View Of Interest) concept as input for path planning algorithms. Game engines could in principle be used for streamlining the production of CH visualizations, but (a) their authoring tools are proprietary and not tailored to CH users, (b) they were not developed for displaying high-resolution models with LOD, (c) long- or even mid-term sustainability may be a problem, (d) content production typically requires programming, and (e) they still require using, e.g., Maya for 3D modeling the scene. Of course, all of these obstacles can be overcome with some development effort. But the key is finding the right concepts. Our (unproven) claim is that also a game engine based authoring tool for CH visualizations will eventually have to share the three key concepts of the system presented in this paper.

Searching for further related work we have found many references related to virtual museums, most notably maybe in the context of the ARCO project [WMD*04, WWWC04], and to exhibition planning [MMPD08, HME*12]. While some of them use similar concepts (e.g., exhibition templates) our work differs in that we aim at virtual exhibits, not virtual exhibitions. We do not want to replace real ('walled') museums, but enhance the understanding appreciation of physical artifacts with virtual ones. We consider the virtual museum literature therefore as only marginally related. To the best of our knowledge, the efficient creation of animated CH visualizations has not been dealt with before.

On the more technical side, the concept of complementary exhibits with enhanced interactivity was proposed by [HSLF07], who emphasized that the artifact rather than technology must be in the focus of the visitor. A sustainable markup concept to integrate additional information using 3D links and anchor points was presented in [HSB*09]. Our system also integrates these concepts, but focuses more on the integration in museum workflows. And as mentioned, our infrastructure approach is adopted from the 3D-COFORM project [Arn09], building upon its 3D asset repository.

3. Curator mode

The curator mainly has four tasks when using our system:

- Getting hold of 3D assets, deciding what is to be shown
- Describing to the designer the required scene templates
- Choosing appropriate scene templates for each 3D kiosk
- Filling templates with 3D assets, configuring the exhibits

Some assets may be acquired by the museum staff themselves (e.g., photo reconstructions of suitable backdrops), but high-quality 3D-acquisition requires trained personnel (e.g., photographic department, scan company). Alternatively, 3D assets can be bought or rented from other museums or companies. The process of finding the right assets for the exhibition is also supported by 3D-COFORM tools. The IVB (Integrated Viewer Browser) provides a visual interface to formulate semantic queries (SPARQL) to the semantic metadata network as described in detail in [DTT*10]. The right assets for the planned exhibition can be found on a semantic rather than purely formal basis, i.e., not just by period or size, but also by style or manufacturing method, if

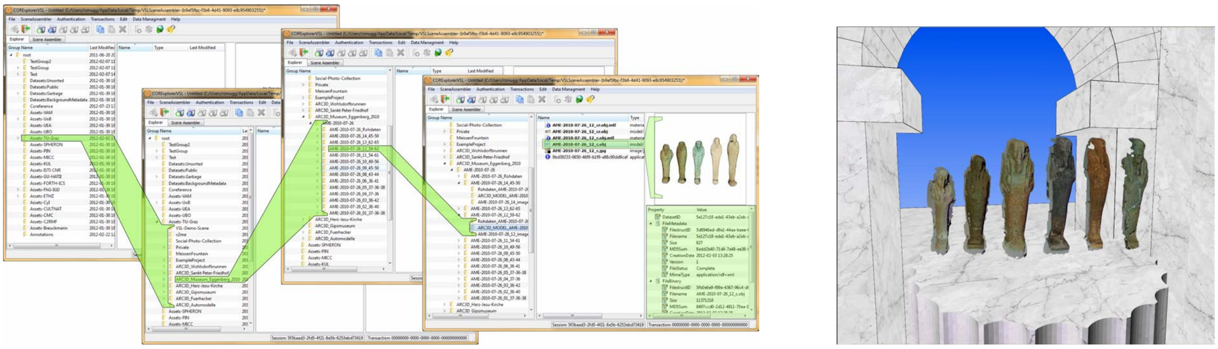

Figure 5: *Curator selecting assets from the RI. CorExplorer allows navigating through the group hierarchy of the distributed asset database. The curator chooses the assets for the exhibits and drags them onto the pre-defined drop targets.*

that information is provided in the metadata. Once a suitable collection is found, it is grouped to facilitate the further steps of asset production using CorExplorer (Fig. 5).

Once it is clear what shall be shown, the curator must decide how to show it: as finding-in-context, as exploding view, as a small animated story, as walkthrough animation, as decomposition into parts, etc. Also more abstract "skins" are possible (grid, carousel, iPod-like cover flow). The curator determines the required functionality and the look and feel in terms of color, layout and design, so that it matches the visual language of the walled exhibition. This description goes to the designer (Section 4) who is responsible for the realization and prepares a set of scene templates for the exhibition.

When the digital assets, 3D artifacts and the scene templates are available, the curator can start creating digital exhibits. For each 3D kiosk in the museum he or she may choose a suitable scene template and insert digital assets via drag-and-drop on pre-defined places, the so-called *drop targets* (Fig. 6). Depending on the skill and requirements, the designer can set certain parameters of the scene to be configuration options changeable by the curator.

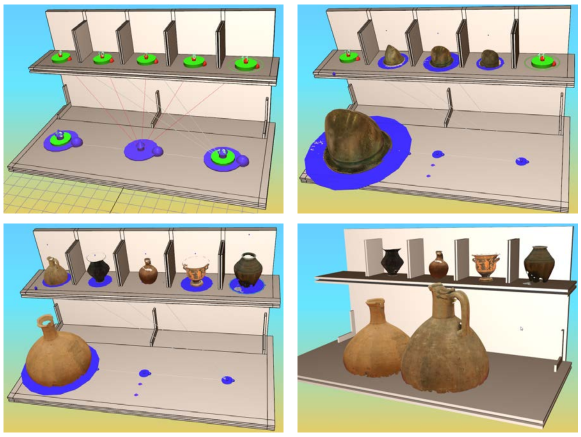

Figure 6: *Curator mode. A scene template for object comparison is re-used. Top: The scene template with drop targets (green) was created by the designer (left). It is filled by the curator via drag & drop with helmets (right). Bottom: It can be re-used with ceramic pots (left); in presentation mode users can select a pot from the shelf for an animated direct comparison of the two shapes (right).*

3.1. Drop targets and scene configuration

The curator mode shall enable curators as non-3D-experts to produce compelling 3D scenes; but scene templates can be simple or more complex. In the simplest case, they contain only a fixed number of drop targets (Fig. 6) on which the curator drags artifacts. Simple scenes can also be filled automatically, e.g., from an Excel sheet. More flexible scene templates may offer more configuration options, giving more freedom and control to the curator, e.g., with rotation widgets to configure the asset orientation, editable animation paths, switching scene parts on or off, etc. Since the scene template is procedural (Section 4) also the number of drop targets may be configurable (Fig. 11). Even with advanced scene templates, curators have to learn only a few key concepts (Fig. 8), e.g., why a scene graph is used (Fig. 7).

3.2. Integration with the repository

We have realized the curator mode as a plugin for CorExplorer, the client software of the Repository Infrastructure (RI) of 3D-COFORM [PBH*10, DTT*10]. The main functionality of CorExplorer is to show the available assets in a directory tree with a preview pane showing image and metadata (Fig. 5). Using the plugin, assets can be directly added to the 3D scene via drag & drop. Since the binary store of the RI is distributed, other staff members can collect suitable sets of assets for the exhibition beforehand and group them in respective directories. Grouping implies no physical copy of the binary data: Since datasets in the RI are immutable,

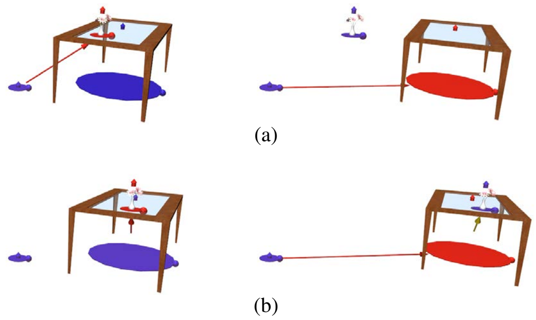

Figure 7: *Advantages of a scene graph. (a) When independent objects move, relative positions change; with larger scenes, many moves are required, and their consistency is difficult to check. (b) With hierarchical scenes, all children (flower) move in sync with their parent node (table); moving a whole sub-tree requires only moving its root node.*

each dataset can belong to any number of groups (*write once policy*). Note that the designer can store the scene templates also in the RI, and the curator can also ingest the scenes created for the kiosks. To avoid chaos, relevant data should not reside on local hard disks.

4. Designer mode

The designer mode gives access to the full set of scene graph and animation operations. Internally, our system is based on an open source scene graph library (OpenSG) with a dataflow engine on top to create the scene graph procedurally on the fly. Technically, this is our main innovation with respect to static scene formats (X3D) since this approach allows much greater parametric changes to the scene structure. Second, our approach is procedural but requires no scripting: The dataflow graph is composed in background from the interactively issued editing operations described below. So all technicalities are hidden away, and also the designer does not have to be a 3D expert. No scripting skills are required for authoring a scene template.

Creating the scene hierarchy. The scene graph concept is easy to grasp by designers (Fig. 7): The relative positioning of one object (child) relative to another (parent) results naturally in a hierarchy of transformations (position + orientation). Web designers know it as the concept of grouping; but in a scene graph, grouped objects can still move relatively. The transformation hierarchy is visible best in animations.

Scene graph nodes are represented graphically as widgets (Figure 8(a)) with three parts: The cylinder for moving parallel to the ground, the arrow for lifting, and the ball for rotation. During motion, arrows indicate the hierarchy, i.e., the parent node and the (direct) child nodes (Fig. 8(e)). When lifting a node, it snaps to the parent level, and to the uppermost plane of the parent (object), to make stacking easier.

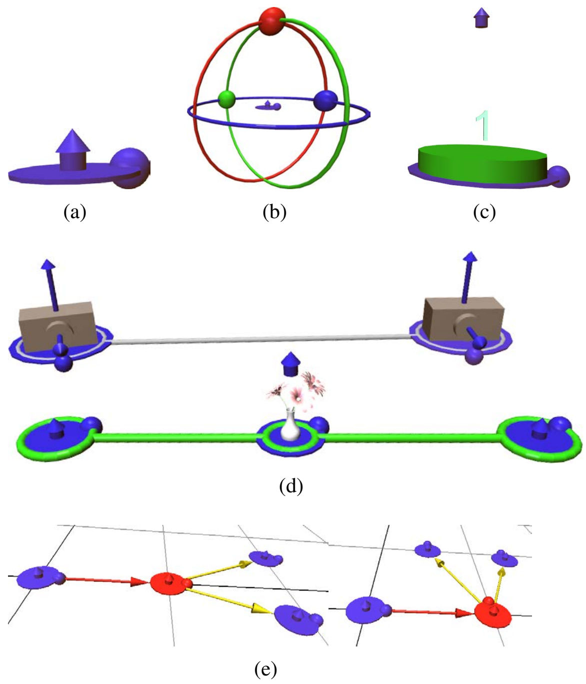

Figure 8: *The five widgets used for scene manipulation: (a) Scene graph node, (b) full rotation widget, (c) drop target, (d) pose interpolation (animation of objects and camera) (e) scene graph hierarchy. Understanding these five concepts (plus time) is sufficient for designing scene templates.*

Integrated 3D modeling engine. We have integrated also a shape modeling engine that works in a push/pull manner like SketchUp but is also parametric [ZKT*12]. It is based on the same dataflow engine as the scene graph, and it produces OpenSG compatible geometry. It allows creating parametric shapes as backdrops (Fig. 10) whose shape parameters can also be configured by the curator. It can be further extended with libraries of scripted assets for windows, arches etc.

Placing objects. One 3D object can be attached to every scene graph node, which affects neither the hierarchy nor the functionality of the node; the widget just resizes according to the size of the object. Building a small scene is shown in Fig. 9. Because we had no models at all for the backdrop, we decided to use, despite the style mismatch, furniture assets from Google Warehouse, since they can be freely used.

Placing drop targets. A second possible attachment to a scene graph node is a numbered drop target (Fig. 8(c)). It marks a spot for the curator to place assets from the repository. Drop targets can also be created parametrically as part of a procedural scene graph, so that their exact number is configurable in curator mode. Different patterns can be constructed, e.g., by using lines and circles as shown in Fig. 11.

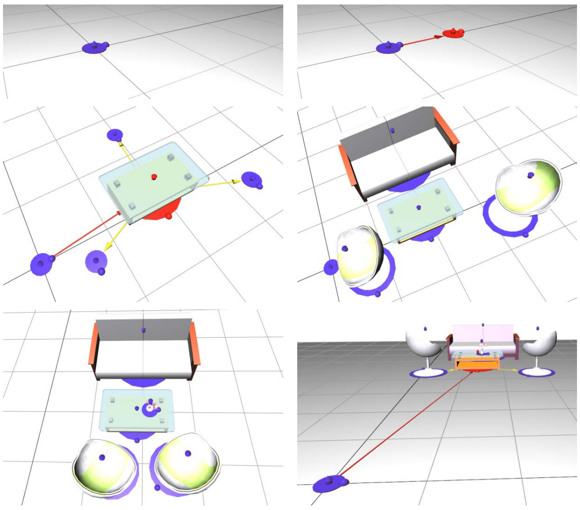

Figure 9: *Widget-based scene creation. By unfolding scene graph nodes (top) the hierarchy (middle) is quickly created and filled (mid right). Relative motion of the parts (bottom left) and of the group (right) are possible.*

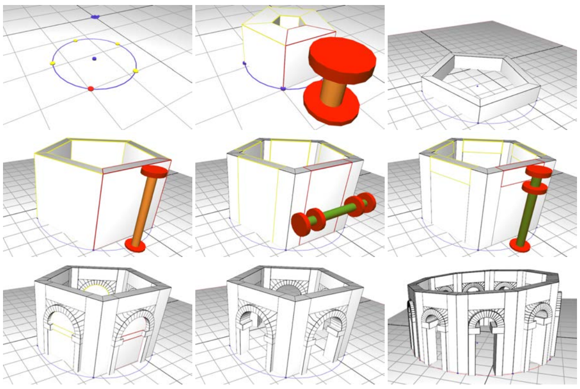

Figure 10: *Integrated parametric modeling engine. It uses volumetric geometric primitives (convex polyhedra) that can be refined and partitioned using plane splits. The usage is similar to the push/pull method of SketchUp. All shape operations are recorded in the dataflow graph and are therefore parametric. Scripted assets like arches and windows are imported from an asset library.*

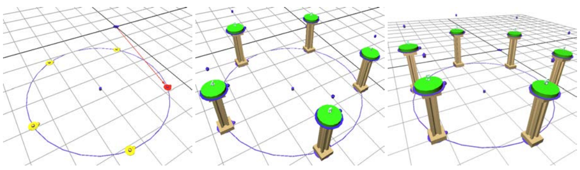

Figure 11: *Procedurally generated drop targets. The specific number of drop targets can be configured later in curator mode. The n-gon is from a compass-and-ruler library.*

Camera placement. Cameras represent pre-defined views that permit directing the attention of the visitor. Cameras can be fixed or remain configurable by the curator who can then, e.g., highlight object details that are complemented by some textual explanations. Cameras are normal scene graph objects, so they can also be defined relative to a (parent) object that remains in view even when it moves (Fig. 12).

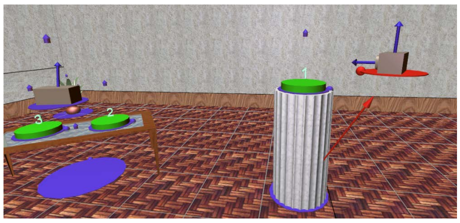

Figure 12: *Camera placement: Pre-defined views. The camera (right) is a child of the pedestal node (red arrow). So the object on the pedestal (drop target 1) remains in view even when the pedestal moves in design mode (or in curator mode, if configured as movable).*

Defining animations. Animations are defined using keyframes. The pose (position + orientation) at the keyframes is defined using scene graph nodes decorated with motion path widgets (Fig. 8(d)). Since cameras are normal scene graph objects, this allows for smooth view transitions. For flight-through animations the system provides camera splines with the view in flight direction. The control points can be edited in curator mode (see Fig. 13).

Defining the functionality of the curator mode. The designer can change many things in the scene that the curator would not want to touch anymore. Larger scenes have a confusing number of options and parameters. Limiting the configuration options not only shields the curator from accidentally doing harm, but also makes the work of the curator simpler, more targeted and efficient. The technical basis for this configurability is the central dataflow graph that contains all scene parameters. In essence we have just added a boolean attachment *curator changeable* to all the values in the dataflow graph. This explains the wide range of possibilities for the configuration of scene templates (object position, camera orientation, length of animation, etc.).

4.1. Presentation Mode and User Interaction

The final step is the upload to a 3D kiosk and the presentation to museum visitors. We provide a standalone viewer application for fullscreen viewing that works also for tiled displays. And like VirtualInspector [CPCS08], our viewer is available as Qt Widget, so it can be used with any Qt application, and also be embedded on a HTML page in a Qt-enabled web browser (WebKit, which is part of Qt). As explained in Section 2, this is extremely helpful in order to streamline

Figure 13: *Fly-through animations along editable splines. The animation was developed using a modern architectural site (Frankfurt fair) with inserted scanned assets. Since the spline is editable in curator mode (bottom left), the scene template could quickly be adopted when the medieval town model (generated by CityEngine) became available.*

content production, since the viewer can be completely configured in a simple way without programming, only by using HTML/QML. Many tools exist for HTML design, and for integrating additional content (text, images, videos) to complement the 3D view. Many normal web designers have the necessary skills, so this requires no specialized staff.

Concerning user interaction we follow a generic approach: Besides mouse and keyboard, user events can be triggered using small command strings sent via a socket connection. This makes it possible to use a variety of input devices for the kiosk, even gesture recognition (Kinect). This flexibility is necessary since mouse and keyboard are sometimes not usable in museums because of health concerns.

5. Results and Discussion

For testing our system we have realized three use cases: Animated fragment reassembly with alternative assembly hypotheses as an example of a custom-made animation for high quality content (Fig. 14); a re-usable scene template for fly-through animations, used with two different models (Fig. 13); and finally the comparison use case with two similar objects that are brought to the same pose to highlight the difference (Figs. 6 and 15). Another scene template with a more 'floating' visual style is shown in the teaser (Fig. 1).

Figure 14: *The Meissen Fountain. The five parts can be arranged in two possible ways (top left, top right). An animation can show an interesting transition between both hypotheses, as well as close-up views of the faces.*

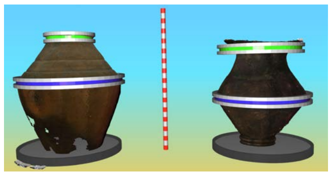

Figure 15: *Comparison of proportions. The girth measurement rings in the scene template are curator configurable to adapt them to the specific imported 3D vase models.*

5.1. Limitations of the current approach

We are aware that the visual quality of our results is definitely not yet up to par with the requirements of a museum presentation. The focus of our paper is on the concepts; in the lack of a laser scanner, our 3D artefacts are acquired from images by photometric stereo using the Arc3D webservice [VG06]. With high-quality artefacts and assets, and using shader materials for the modeled geometry, we can certainly obtain the same visual quality as, e.g., the Unity3D viewer (Fig. 2), provided design-oriented people use the tools, rather than engineers or archeologists.

There is also room for improvement with respect to usability. The visual representation of the scene graph using connecting lines can be cluttered for larger scenes; the experiments suggest that a bounding-box based grouping metaphor might be easier to control, which is what we will try next. The handling of time can be improved as well. With many things moving, it is sometimes not easy to avoid collisions at a particular time. This could be checked automatically to insert new keyframes.

6. Conclusion and Future Work

We have presented concepts and a prototype of an integrated system for the efficient creation of interactive museum exhibits. It addresses a difficult and important problem: Enabling curators as non-3D-experts to create complex animated 3D scenes as museum exhibits. Curators are experts in visual communication, and for 3D to become a standard in museums leveraging on their expertise is a key issue.

We are convinced that any system for the streamlined production of animated CH visualizations - and we hope there will be many - will have to use the same concepts as ours, namely a reduced set of scene graph and shape modeling operations, direct repository integration, and the separation of designer from curator mode by using scene templates.

There are many opportunities for further research. As mentioned, we must improve the visual quality and the usability of the system. We will continue and intensify the collaboration with CH professionals in order to produce convincing showcases and best-practice examples. Animated scene graphs are a very generic and versatile method for CH visualizations. It will be most interesting to see how this new tool is used by CH professionals to show things they could not easily, or not at all, show before.

Another big challenge is sustainability. We have to find a description standard for the filled scene templates (dataflow graph) and metadata (semantic description of the exhibit). We envisage that good scene templates will be often re-used and exchanged between exhibitions.

7. Acknowledgements

We gratefully acknowledge funding by the European Union for the 3D-COFORM project (grant FP7 ICT 231809). We wish to thank Thomas Reuter and Rengert Elburg from the Archaeological Heritage Office in Saxony, Germany, for the images of the prehistoric well in Altscherbitz. The 3D animation in Unity3D was created by Marc Proesmans from KU Leuven; it is available from the 3D-COFORM website.

References

[Arn09] ARNOLD D.: Tools and Expertise for 3D Collection Formation. In *Proceedings of EVA 2009 Florence* (2009), pp. 94–99. 3

[BSly] BBC, SILICON GRAPHICS (SGI): Mummy: Inside story, 2004 July. Exhibition at the British Museum London, http://www.thebritishmuseum.ac.uk. 3

[CIR*10] CHITTARO L., IERONUTTI L., RANON R., VISINTINI D., SIOTTO E.: A high-level tool for curators of 3d virtual visits and its application to a virtual exhibition of renaissance frescoes. In *Proc. VAST 2010* (Paris, 2010), Eurographics/Blackwell Publishing, pp. 147–154. 3

[CPCS06] CALLIERI M., PONCHIO F., CIGNONI P., SCOPIGNO R.: Easy access to huge 3d models of works of art. In *Fourth Eurographics Italian Chapter 2006* (2006), Fellner D., (Ed.), Eurographics Association, pp. 29–36. Catania, 22-24 Feb. 2006. 3

[CPCS08] CALLIERI M., PONCHIO F., CIGNONI P., SCOPIGNO R.: Virtual inspector: a flexible visualizer for dense 3d scanned models. *IEEE Computer Graphics and Applications 28*, 1 (Jan.-Febr. 2008), 44–55. 3, 6

[DTT*10] DOERR M., TZOMPANAKI K., THEODORIDOU M., GEORGIS C., AXARIDOU A., HAVEMANN S.: A Repository for 3D Model Production and Interpretation in Culture and Beyond. In *Proc. VAST 2010* (2010), pp. 97–104. 3, 4

[HME*12] HECHER M., MÖSTL R., EGGELING E., DERLER C., FELLNER D. W.: 'Tangible Culture' – Designing Virtual Exhibitions on Multi-Touch Devices. In *Social Shaping of Digital Publishing, ELPUB 2012* (2012), Baptista A. A., Linde P., Lavesson N., de Brito M. A., (Eds.), IOS Press, pp. 104–113. 3

[HSB*09] HAVEMANN S., SETTGAST V., BERNDT R., EIDE Ø., FELLNER D. W.: The Arrigo Showcase reloaded – towards a sustainable link between 3D and semantics. *ACM Journal on Computing and Cultural Heritage (JOCCH) 2*, 1 (2009), 1–13. doi: http://dx.doi.org/10.1145/1551676.1551680. 3

[HSLF07] HAVEMANN S., SETTGAST V., LANCELLE M., FELLNER D. W.: 3D-Powerpoint – a design tool for digital exhibitions of cultural artifacts. In *Proc. VAST 2007 Intl. Symp.* (2007), Eurographics, pp. 39–46. doi: 10.2312/VAST/VAST07/039-046. 3

[MMPD08] MURG S., MORITSCH O., PENSOLD W., DERLER C.: Using standardized methods to present three-dimensional content on the web in the context of cultural heritage. In *Museums and the Web 2008* (Toronto, March 2008), Trant J., Bearman D., (Eds.), Archives & Museum Informatics. 3

[MSLV08] MATEEVITSI V., SFAKIANOS M., LEPOURAS G., VASSILAKIS C.: A game-engine based virtual museum authoring and presentation system. In *Proc. Digital Interactive Media in Entertainment and Arts* (New York, NY, USA, 2008), DIMEA '08, ACM, pp. 451–457. 3

[PBH*10] PAN X., BECKMANN P., HAVEMANN S., TZOMPANAKI K., DOERR M., FELLNER D. W.: A distributed object repository for cultural heritage. In *Proc. VAST 2010* (2010), pp. 105–114. doi:10.2312/VAST/VAST10/105-114. 4

[TSP*12] TRAPP M., SEMMO A., POKORSKI R., HERRMANN C.-D., DÖLLNER J., EICHHORN M., HEINZELMANN M.: Colonia 3d - communication of virtual 3d reconstructions in public spaces. *International Journal of Heritage in the Digital Era (IJHDE) 1*, 1 (2012), 45–74. 3

[VG06] VERGAUWEN M., GOOL L. V.: Web-based 3d reconstruction service. *Machine Vision Applications*, 17 (2006), 411–426. 7

[WMD*04] WHITE M., MOURKOUSSIS N., DARCY J., PETRIDIS P., LIAROKAPIS F., LISTER P., WALCZAK K., WOJCIECHOWSKI R., CELLARY W., CHMIELEWSKI J., STAWNIAK M., WIZA W., PATEL M., STEVENSON J., MANLEY J., GIORGINI F., SAYD P., GASPARD F.: Arco " an architecture for digitization, management and presentation of virtual exhibitions. In *Proc. Computer Graphics International* (Washington, DC, USA, 2004), CGI '04, IEEE Computer Society, pp. 622–625. 3

[WWWC04] WOJCIECHOWSKI R., WALCZAK K., WHITE M., CELLARY W.: Building virtual and augmented reality museum exhibitions. In *Proceedings of the ninth international conference on 3D Web technology* (New York, NY, USA, 2004), Web3D '04, ACM, pp. 135–144. 3

[ZKT*12] ZMUGG R., KRISPEL U., THALLER W., HAVEMANN S., PSZEIDA M., FELLNER D. W.: A new approach for interactive procedural modelling in cultural heritage. In *Proceedings of the 40th Conference of Computer Applications and Quantitative Methods in Archaeology (to appear)* (2012). 5