EG UK Theory and Practice of Computer Graphics (2010)
John Collomosse, Ian Grimstead (Editors)

Computer Modelling of Theatrical Sets

Ian Stephenson[†1] and Rebecca Pride[‡2]

[1]National Centre for Computer Animation,Bournemouth University,UK
[2]Arts University College, Bournemouth, UK

Abstract

Here we discuss the design and implementation of a modelling tool specifically targeted for use by designers of theatrical sets. By applying domain specific knowledge and contraints to the design of the software, we demonstrate a system which can be used without signifigant training, by users with no previous computer graphics experience. We present initial end user experiences of using the tool in designing a real production of "The First Arabian Night".

Categories and Subject Descriptors (according to ACM CCS): I.3.8 [Computer Graphics]: Applications

1. Background

A number of systems exist which use computer graphics to visualise stage sets. However from early research papers [DAG95], through to current commercial software such as "Capture", "SunLite" or "ESP Vision" these have primarily concentrated upon the rendering of lighting effects. Before the set can be rendered in these packages it must first be modelled.

The facilities for designing the stage set within these packages is either limited to general layout, and positioning of lights for concert style shows, or absent completely. It is typically recommended that the user produce a model of their set in a third party modelling package: Maya and 3D Studio Max are often used, while for larger productions CAD solutions such as AutoCAD are also appropriate. However the investment required (both financial, and in terms of the users' time) by the modelling package is likely to far exceed that of the visualisation and rendering tools.

One of the few projects to tackle the problem of modelling sets is OpenStages [Dye00] (Figure 1a). Developed by a set designer, this package benefits from the author's considerable theatrical experience. It provides facilities specifically tailored to the problem of theatrical sets, using terminology which designers are familiar with. However the ageing program shows its limitations in other respects. It's interface design is overly complex, while many features are only basically implemented, or implemented inconsistent ways. The program also has a number of limitation imposed by it's limited use of OpenGL — Transparency is handled in a confusing way, and only 7 lights my be used at once. As a result of these problems it has not been widely adopted.

Our previous work [Ste09] is image based, and uses photographs of a set (or model). While this provides valuable flexibility for testing lighting once a set has been built, it cannot help at the earlier design stages, and cannot show the scene from different perspectives.

Lightweight modelling systems such as Google SketchUp are attracting interest in the scenography community, though it is better suited to the design of props than theatrical sets. In particular SketchUp's lack of lighting support is a major limitation. When creating a new scene the user is presented with an origin placed on an infinite plane which fails to create any sense of a space that the user can navigate through, and in general it is too easy for the naive user to become confused when they system behaves in ways they didn't expect (though SketchUp performs far better in this regard than most other 3D apps).

Specialised software exist for applications such as kitchen design (Figure 1b), allowing operators to assemble and configure plans and visualisations of room layouts quickly and

† e-mail: ian@dctsystems.co.uk
‡ email: rpride@aucb.ac.uk

DOI: 10.2312/LocalChapterEvents/TPCG/TPCG10/075-081

Figure 1: *Two Domain Specific Modelling Systems: a) OpenStages b) MicroSpot Interiors*

easily. By incorporating domain knowledge and restrictions into the software, the user is guided more quickly towards valid solutions. If such an approach could be adapted to stage design then sets could created and modified without extensive computer graphics training.

Existing working scenographic techniques are predominantly paper based, and 3D visualisation is by the construction of 1:25 scale models. A suitable computer modelling tool should support designers at all stages of their workflow, from early concept work, through to rehearsals (where directors may use models to test and explain scenes to actors).

In general there is considerable interest in the use of computer modelling among scenographers, but they are essentially a non-technical group. The learning curve and overall complexity of most modelling and CAD software prohibits its adoption, and in order to bring computer visualisation to this audience, tools must be developed which are simple to use, and targeted to the requirements of specific field.

2. Software Design

From our experience developing lighting simulations, we have identified the need for modelling software which could support our own work, and the work of set designers. To achieve this a number of basic principles were established:

- Only include essential functionality for theatre
- Use of theatrical and scale modelling rather than graphics terminology
- Prebuilt props
- Reduced degrees of freedom on positioning
- Strict, "Intelligent" parenting
- Create the sense of a real space

The system, known as StageBuilder is implemented on MacOSX using the Cocoa NSDocument architecture, which ensures the app has a standard Mac "look and feel" to minimise user learning curve. It was our hope that a set designers with no computer graphics experience could be productive on the system within a few hours.

2.1. Basic Interface

The system provides three kinds of graphical views of the system: Orthographic plan and elevation views, and a 3D

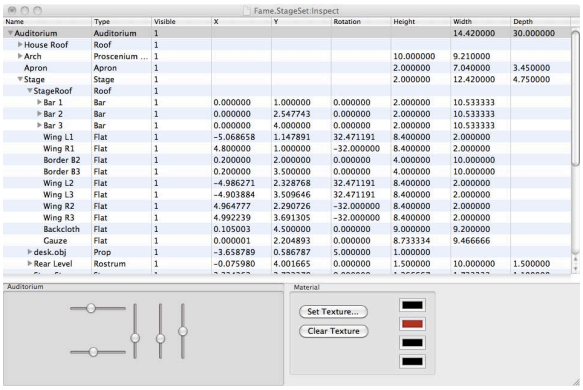

Figure 3: *StageBuilder's Tabular "Inspector" View*

perspective view which the user can move around in (Figure 2). The user can select objects in each view, move them around and rotate them. There are no shading options, as such features are unnecessary — the 3D view is always fully lit and shaded, while the 2D views are always wireframe. When a new scene is created by default it has a 3D and a plan view, but any number and combination of views can be created.

In addition to these, a tabular view of the scene is displayed in another window (Figure 3). Though this was initially provided as a more "technical" feature for entering key dimensions, it quickly became apparent that, due to their background experience of model making and the accuracy which that requires, set designers are very comfortable dealing the exact numerical positioning and dimensions of the objects in their scene. While direct manipulation was available in all the graphical views, along with a GUI interface for more specific object parameters, users gravitated towards editing the numerical data directly.

2.2. Stage Props

Every scene automatically contains the basic components of a Proscenium style theatre [Hog75]:

- An Auditorium (with roof)
- A Proscenium Arch
- A Stage (with roof)
- An Apron
- A Basic House Light

The default scene is shown in figure 4. The dimension of each object in the scene can be adjusted to quickly match most theatres, including thrust and studio layouts.

To this basic theatre the user can add standard primitives:

- Rostra
- Ramps
- Steps

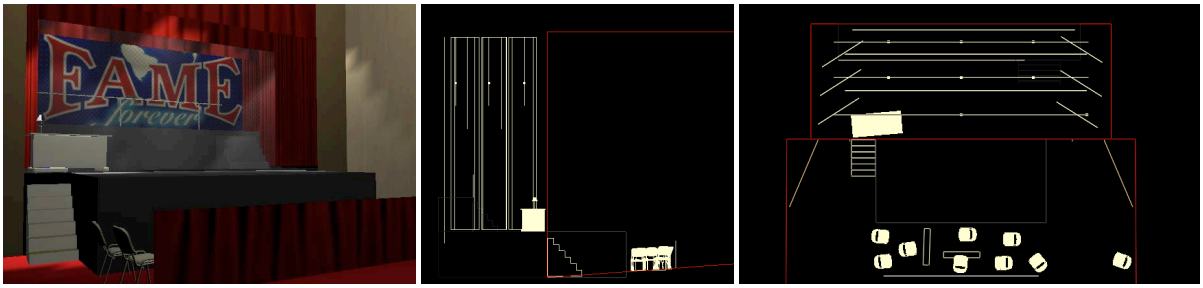

Figure 2: *Interactive Views Within StageBuilder: 3D, Elevation and Plan*

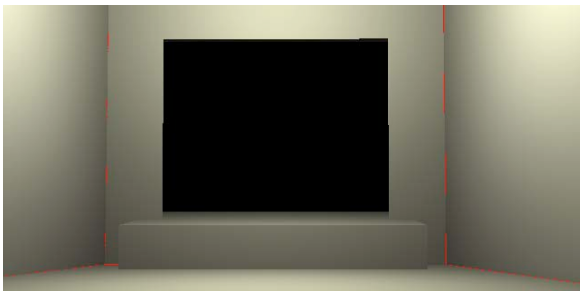

Figure 4: *StageBuilder's Default Proscenium Arch Scene*

- Flats
- Lighting Bars
- Lamps.

These are the basic components of stage sets, and by positioning, combining and adjusting these a rough set can be quickly constructed. All of these objects have simple texture coordinates predefined, and the user can simply attach an image file to customise the appearance of an object. The alpha channel can by used to create complex cutouts in the simply shaped base objects.

There is no provision for free form modelling within the package. Where specific props are required they are simply imported from obj files. The user has the option to model custom objects in an existing modelling packages: SketchUp being suited to novice users, while more experienced users are likely have their own preferences. More typically however the user can purchase and/or download an existing model. This avoids overloading the user with tools they do not need, and uses existing tools to perform a task that is already well researched and supported.

There is established theatrical practise of a "Prop Library", and while users where initially confused by loading models, when the UI as adjusted so that they simply selected a model from their prop store, they were much more comfortable, even though the underlying functionality was unchanged. Scenographers tend to rely on a number of common props, so while they initially may require support from an experienced 3D poly modeller to develop their library of props, they quickly acquire the objects they use most frequently, and become increasingly independent.

2.3. Parenting

One of the core design concepts of StageBuilder is that all objects are parented by attaching them to another object (though explicit use of the term "parent" is avoided). While this principle is common to most complex modelling and animation packages, StageBuilder enforces this more strictly by insisting that all objects are "on" their parent object.

Rather than being able to position an object freely in 3D space (with six degrees of freedom) the child objects vertical position is established by the height of its parent object. In addition objects stand upright, with only one axis of rotation, restricting them to three degrees of freedom. While this is far less flexible, it mirrors the way users thing about placing objects in the real world, and enables realistic models to be constructed quickly.

This concept is extended to the 3D camera — rather than providing the free moving camera found in most 3D packages, the viewer is considered to be a person standing or sitting within the auditorium. They are free to move around, but their height is restricted to 1.5m above the surface they are standing on. This simplifies camera positioning considerably, and increases the feeling of being "in" the virtual space. Support for a dual joystick type game controller enables the most naive users to explore the set without prior training.

The system deals intelligently with suspended objects, without requiring any special consideration by the user. The stage and house roofs, and the proscenium are hard coded to be suspension points, and objects are automatically "hung" rather than placed upon objects which are marked as such. Objects which are themselves hanging inherit their parents status as suspension points.

2.4. Lighting

As with other objects, lights have a greater physical presence than in most modelling packages. They are placed in the

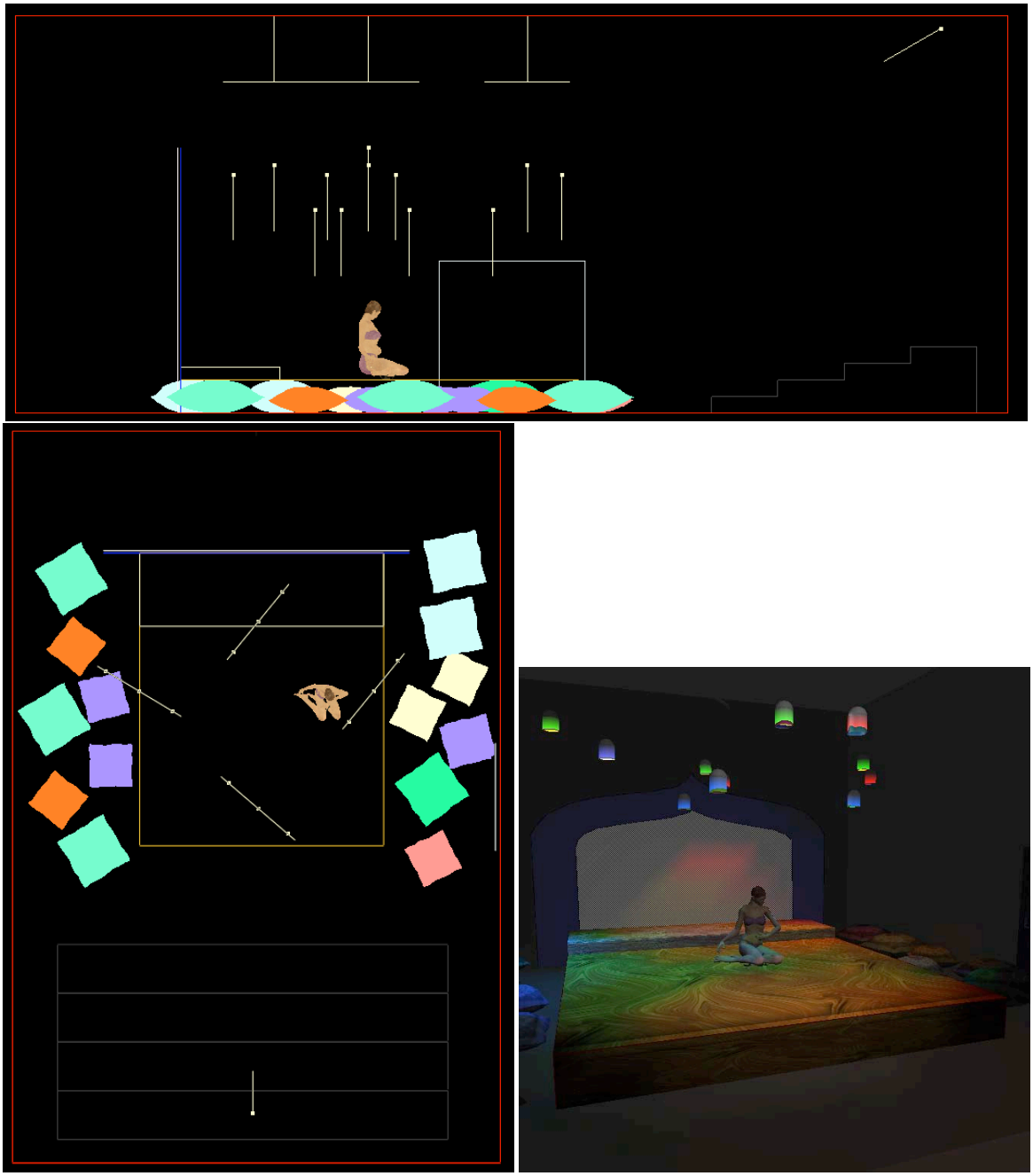

Figure 5: *Views of the Arabian Nights Stage Set, Visualised within StageBuilder*

scene like any other object, and as such must be attached to another object which supports them, rather than being freely placed in 3D space. They are represented on screen as a simple 3D model, which is (unlike other modelling systems), is included in final renders — real world lights are not invisible.

OpenGL supports only 8 lights in a scene, but theatrical sets typically use dozens of lights simultaneously. The technique used in games to overcome this limitation is to select the 8 most important lights for each object in the scene, and light each object individually, the assumption being that the more distant and less intense lights to not make a significant contribution. However in theatrical lighting the illumination for each object is often made up of many small contributions from a large number of lights, and in many cases all lights could be essential to produce the sense of theatricality which users want in their designs.

Instead the scene is rendered multiple times, with 8 lights active in each pass. These are summed in the accumulation buffer. This has the additional benefit that the results can be gamma corrected when they are transferred back to the primary buffer.

Lights are allocated DMX [ANS04] channels, and their intensity can be controlled from a DMX lighting desk via a DMX/USB interface. However it is not the intention that this application provide accurate lighting simulation — rather it should export the modelled scene to a lighting visualisation system.

3. Application

The system was used in the design of the set for a production of "The First Arabian Night" [Sha75] to be directed by Sean Aita. No artwork was produced prior to building the set on the computer.

The production was to be set in a studio theatre, rather than the Proscenium theatre to which the application defaults, but by hiding some of the default stage components, and adjusting their dimensions it was possible to quickly reproduce the performance space, including audience seating, access doors and fire escapes. The resulting sense of place was noted as particularly valuable.

This ability to place the director and other production staff "in the space" during the earliest parts of the design phase was considered one of the most exciting facilities. Though the scenographer's lack of computer experience meant that she was uncomfortable demonstrating the set as a 3D model within the application, the rendered images which were exported, were found to be particularly valuable in meetings with the director, and were frequently emailed and posted online. Examples of these are shown in figure 5. High quality renders where produced by exporting RenderMan RIB files [Ups89]. These were rendered using the Angel [Ste04] renderer and are shown in figure 6.

It quickly became clear from observing the designer, that while set designers are essentially non-technical, they are particularly comfortable working with scale and dimension. As such, while the visualisation provided by the 3D view was invaluable as a communication tool, they gravitated to the more precise plan and elevation views, and having roughly positioned objects, they invariably fine tuned them using the numerical entry in the inspector.

The stylised design of the "Arabian Night" set required only a small number of props, which were easily obtainable from online sources. The cushions used a single model which was textured and coloured to create variation. The headboard was constructed as flattage with a painted texture used to create the shape. A standard female character was placed on the bed for scale. The only significant part of the set which could not be easily constructed was an "enormous duvet" required to cover the "bed". This obviously non-standard object could have been custom modelled and imported, but in practise the scenographer (and director) where happy to simply texture the rostrum.

While the scenographer was able to quickly grasp the concepts of the application, the biggest problems faced were in simply general computing tasks — managing files, scanning and editing images to use as textures, general window management, and mouse/keyboard interaction (which while standard UI practise is more advanced than found in basic office applications).

Within the application, lighting was considered the most difficult aspect to use, even though the lighting controls are considerable simplified compared to the potential complexity of real lighting. While there is scope for improving the application in this area, it re-enforces our experience that set designers and lighting designers have very different skill sets. However it users noted that being able to see their designs lit with even basic theatrical illumination was one of the most rewarding aspects of using StageBuilder.

Having developed the set for one production using the application, the scenographer and her colleagues were very enthusiastic to apply the system on a wider range of productions, including a proscenium based production [PHS10].

4. Conclusions

The system allowed stage sets to be constructed quickly, and potential set ideas tested. It was used in the development of a production of "The First Arabian Night". While this set was simple, and didn't fully stretch the system, it demonstrated its potential. New users were able to grasp the operation of the system, though they did struggle with general computing tasks, such as scanning, and painting images. With little training they were able to develop, visualise and communicate their ideas.

Scenographers quickly identified a role for the application

within their workflow, as a bridge between quick concept and layout sketches, and the construction of scale models. Though they are unlikely to stop using physical models in the immediate future, the ability to communicate the feeling of the performance space to the director before proceeding to model-making, was seen as a great step forwards.

While in principle general modelling packages (such as Maya) incorporate all of the functionality found within the StageBuilder application, it is important to note that they are not suitable tools for designing stage sets. By targeting modelling applications to specific problem domains the modelling process can be speeded up, simplified, and domain specific constraints enforced. Simply adopting domain specific terminology can be significant. While general purpose modelling packages attempt to be applicable to any field, in their generality they exclude all but the computer modelling specialist.

References

[ANS04] ANSI: *E1.11 - USITT DMX512-A Asynchronous Serial Digital Data Transmission Standard for Controlling Lighting Equipment and Accessories*, 2004. 5

[DAG95] DORSEY J., ARVO J., GREENBERG D.: Interactive design of complex time-dependent lighting. *IEEE Comput. Graph. Appl. 15*, 2 (1995), 26–36. 1

[Dye00] DYER C.: Openstages. http://openstages.co.uk/, 2000. 1

[Hog75] HOGGETT C.: *Stage Crafts*. Adam & Charles Black, London, 1975. 2

[PHS10] PRIDE R., HARGREAVES W., STEPHENSON I.: Stagebuilder: a tool to create effective workflow? In *Theatre and Performance Research Association Conference* (2010). 5

[Sha75] SHARED EXPERIENCE THEATRE COMPANY: *The first Arabian Night: The loves of Kamar and Budur*, 1975. 5

[Ste04] STEPHENSON I. (Ed.): *Production Rendering*. SpringerVerlag, 2004. 5

[Ste09] STEPHENSON I.: Digital relighting for stage use. In *EG UK Theory and Practice of Computer Graphics 2009*, Tang W., Collomosse J., (Eds.). Eurographics Association, Cardiff, UK, July 2009. 1

[Ups89] UPSTILL S.: *The RenderMan Companion*. Prentice Hall, 1989. 5

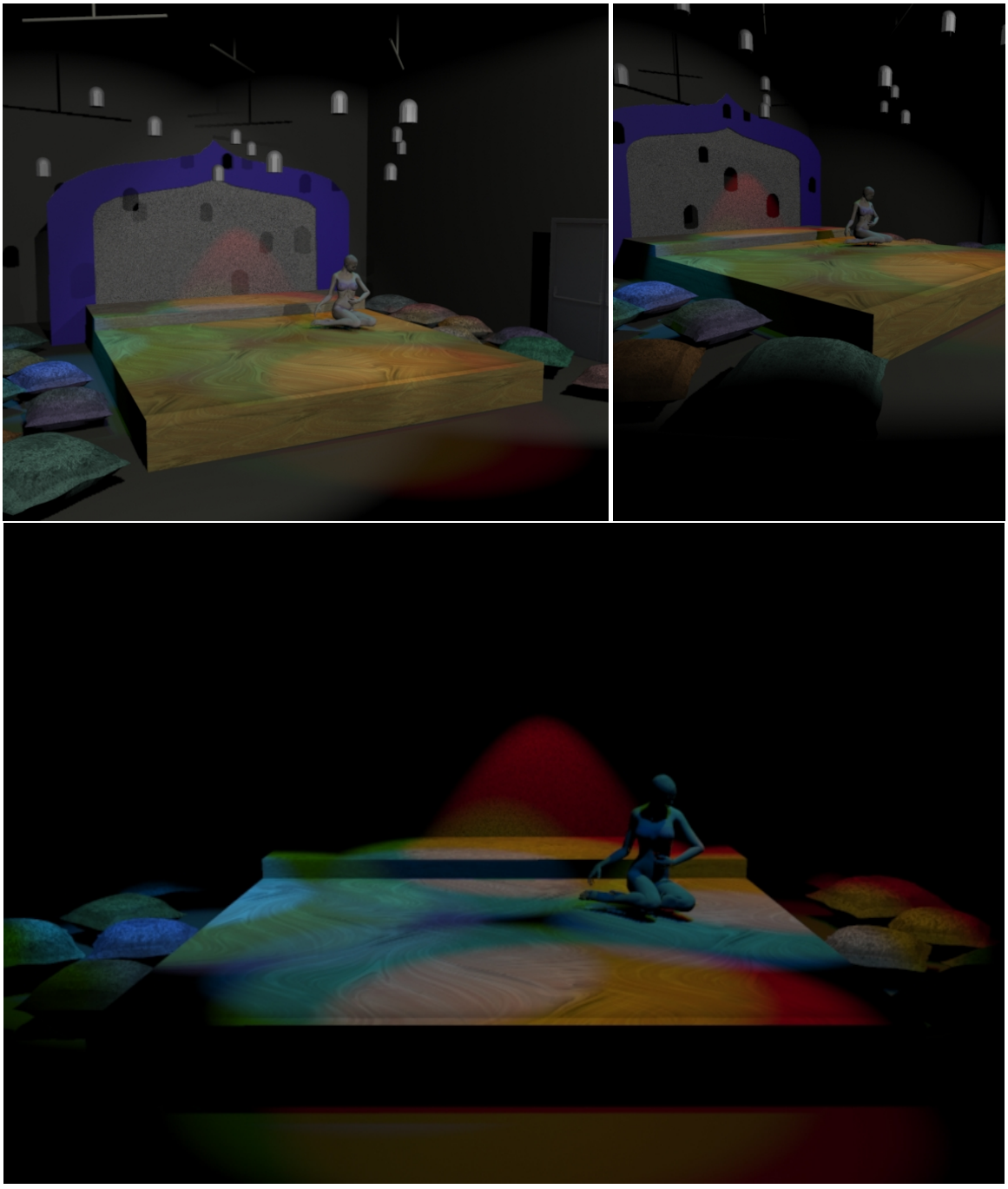

Figure 6: *The Arabian Nights Stage Set, Exported and Rendered in Angel*