

Jürgen Rienow

## Echtzeitvisualisierung mit digitaler Kuppelprojektion

2011

<https://doi.org/10.25969/mediarep/18227>

Veröffentlichungsversion / published version  
Sammelbandbeitrag / collection article

### Empfohlene Zitierung / Suggested Citation:

Rienow, Jürgen: Echtzeitvisualisierung mit digitaler Kuppelprojektion. Marburg: Schüren 2011 (Jahrbuch immersiver Medien 3), S. 77–87. DOI: <https://doi.org/10.25969/mediarep/18227>.

### Erstmalig hier erschienen / Initial publication here:

<https://link.iue.fh-kiel.de/index.php/2011/01/01/jahrbuch-immersiver-medien-2011-online-immersion-annaeherung-abgrenzung-erkundung/>

### Nutzungsbedingungen:

Dieser Text wird unter einer Creative Commons - Namensnennung - Weitergabe unter gleichen Bedingungen 4.0/ Lizenz zur Verfügung gestellt. Nähere Auskünfte zu dieser Lizenz finden Sie hier:

<https://creativecommons.org/licenses/by-sa/4.0/>

### Terms of use:

This document is made available under a creative commons - Attribution - Share Alike 4.0/ License. For more information see:

<https://creativecommons.org/licenses/by-sa/4.0/>

# ECHTZEITVISUALISIERUNG MIT DIGITALER KUPPELPROJEKTION

Jürgen Rienow

## Zusammenfassung/Abstract

Während die Hersteller von Visualisierungssystemen für digitale Kuppelsysteme immer mehr Echtzeitmöglichkeiten in ihre Software integrieren, nutzen die Produzenten von Kuppel-Shows diese Möglichkeiten in der Regel nicht, sondern produzieren kuppelführendes Video. Dieser Artikel untersucht die Gründe für diese auf den ersten Blick widersprüchlichen Entwicklungen und zeigt Möglichkeiten auf, wie mehr Echtzeitvisualisierung und Interaktion in Kuppeln realisiert werden kann und wo dabei Probleme auftreten können.

*While the developers of digital dome projection systems include a lot of real time features to their software, the producers of full dome shows do not take advantage of those features and produce full dome video, most of the time. This article discusses reasons for these conflicting developments and shows possibilities for realizing more real time visualization and interaction in domes and some of the problems that may happen during the process.*

Die Visualisierung mit Echtzeitsystemen (vgl. Akenie-Möller/Haines/Hoffman 1999) ist auf dem Markt der digitalen Kuppelprojektionssysteme stark verbreitet, man findet jedoch selten Anwendungen, die einen Echtzeitanteil haben. Echtzeit und vor allem die damit mögliche Interaktion gelten aber als ein wichtiger Bestandteil immersiver Umgebungen. Dies bestätigt auch Oliver Grau in seiner Dissertation: «Medienstrategisches Ziel ist ein möglichst hochgradiges Gefühl der Präsenz – ein Eindruck suggestiver Anwesenheit, der durch die Interaktion [...] noch gesteigert werden kann» (2002: 15).

Dieser Artikel untersucht die Gründe für diesen Mangel an Anwendungen, stellt dafür vorher die Prinzipien von Echtzeitvisualisierung vor und gibt Vorschläge, wie in Zukunft stärker Interaktion und Echtzeitdarstellungen Einzug in Kuppelmedien finden können.

## Prinzipien der Echtzeitvisualisierung

Um die Vorteile von in Echtzeit generierter Bilder in einem Kuppelmedium einordnen zu können, ist zum einen eine Klärung des Begriffes «Echtzeit» nötig, der in verschiedenen Disziplinen unterschiedlich eingesetzt wird. Zum anderen hilft eine Gegenüberstellung der Produktionsmethoden von Echtzeit- und vorgegenerter (filmischer) Visualisierung in Kuppeln.

Der Begriff «Echtzeit» im Sinne dieses Artikels stammt aus der Informatik. Das engl. «real-time» bedeutet lediglich, «dass ein System auf ein Ereignis innerhalb eines vorgegebenen Zeitraums reagieren muss. Der Begriff sagt nichts über die Verarbeitungsleistung eines Systems aus» (Wikipedia 2011a). Buttazzo geht sogar noch weiter und bezieht die Berechnungszeit mit in die Korrektheitsüberlegungen ein: «[...] [The] correct behavior of these systems

depends not only on the value of computation but also on the time at which the results are produced» (2004: 1) und bezieht diese Aussage insbesondere auf sicherheitskritische Systeme.

Bei Visualisierungssystemen ist das Ereignis in der Regel die Aufforderung, ein neues Einzelbild zu errechnen, der vorgegebene Zeitraum ergibt sich aus der Framerate<sup>1</sup>. Zum Beispiel hat das Echtzeitsystem des Mediendoms Kiel eine Framerate von 60 fps, d. h. das Errechnen eines neuen Frames darf nicht länger dauern als 16,67 ms (1/60 Sekunde). In diesem Zeitraum sind für den Computer einige Dinge zu tun, die sich vom Echtzeit-Rendering (vgl. Wikipedia 2011b) auf einem Einzelcomputer unterscheiden: In einer Kuppel gibt es mehrere Projektoren, hinter denen jeweils ein (oder im Falle von hochauflösenden Projektoren auch mehrere) Grafikrechner als Bildzuspieler arbeiten. Da jeder Rechner einen unterschiedlichen Teil der Echtzeitszene ausrechnen muss (an der Kuppel setzt sich durch die Anordnung der Projektoren wieder das Gesamtbild zusammen), ist jeder unterschiedlich schnell fertig, d. h. man bekommt unterschiedliche Frameraten auf jedem Rechner. Diese sind als unterschiedlich starkes Flimmern wahrnehmbar und zu vermeiden. Dies bedeutet, dass jeder Grafikrechner nicht nur seinen Teil des Bildes ausrechnen muss, sondern es erst dann darstellen darf, wenn alle anderen Rechner ebenfalls ihr Bild berechnet haben. Dieser Vorgang erfordert Synchronisation über ein Netzwerk, entweder in einem LAN (ein möglicher Algorithmus wurde bereits 2002 vorgeschlagen; vgl. Rienow 2002) oder mit spezieller Hardware, die *genlock*<sup>2</sup>- oder *framelockfähig*<sup>3</sup> ist. Synchronisation von mehreren Rechnern erfordert Zeit, die beim Berechnen der eigentlichen Bilddaten berücksichtigt werden muss. Ebenfalls erfordert das Synchronisieren über das Netzwerk eine Möglichkeit, mit Berechnungsdauern von mehr als 16,67 ms sinnvoll umgehen zu können (z. B. durch das Weglassen von einzelnen Frames).

Der internationale Markt von sogenannten Fulldome-Shows, also marktreifen kuppelfüllenden Visualisierungen, ist im Moment auf Fulldome-Video (also vorgerendertes Material) beschränkt. Nur sehr selten findet man interaktive Anwendungen, wie z. B. in der Show *ALIEN ACTION* (Ralph Heinsohn & Dominic Bünning, D 2007), die am Mediendom Kiel produziert wurde (vgl. Heinsohn & Bünning 2007: 57).

Die Produktion eines Fulldome-Videos ist ein zeit- und speicherintensiver Prozess. Für heutige Pro-

duktionen findet die Berechnung der Einzelbilder mit 4096 x 4096 Pixel (4k-Auflösung) statt. Diese großen Bilder, die mit 30 fps oder selten auch 60 fps abgespielt werden, müssen nicht nur berechnet, sondern auch gespeichert werden, so dass für eine 45 Minuten lange Fulldome-Show bis zu 1 TB Daten zusammenkommen können.

Das Arbeiten mit einer Echtzeit-Umgebung löst dieses Ressourcen-Problem: durch das Echtzeit-Rendern der Bildinhalte verkürzt sich die Produktionszeit deutlich wegen des Wegfallens der z. T. wochenlangen Renderzeiten, Szenen sind leicht anpassbar und Interaktion wird möglich.

Allerdings ist die visuelle Qualität bei Echtzeitsystemen begrenzt, denn das Rendern der Bilder geschieht unter zeitlicher Begrenzung. Das größte Problem allerdings ist die mangelnde Kompatibilität der Fulldome-Echtzeitsysteme untereinander. Das Spiel aus *ALIEN ACTION* funktioniert beispielsweise nur auf anderen Kuppeln mit einem *Digistar3*-System.

## Echtzeit-Inhalte

In der Regel werden bei einem Fulldome-System viele Objekte und Funktionalität mitgeliefert, die aus den Anforderungen eines Planetariums motiviert sind, denn heute sind die einzigen Fulldome-Spielstätten (mit wenigen Ausnahmen) Planetarien.

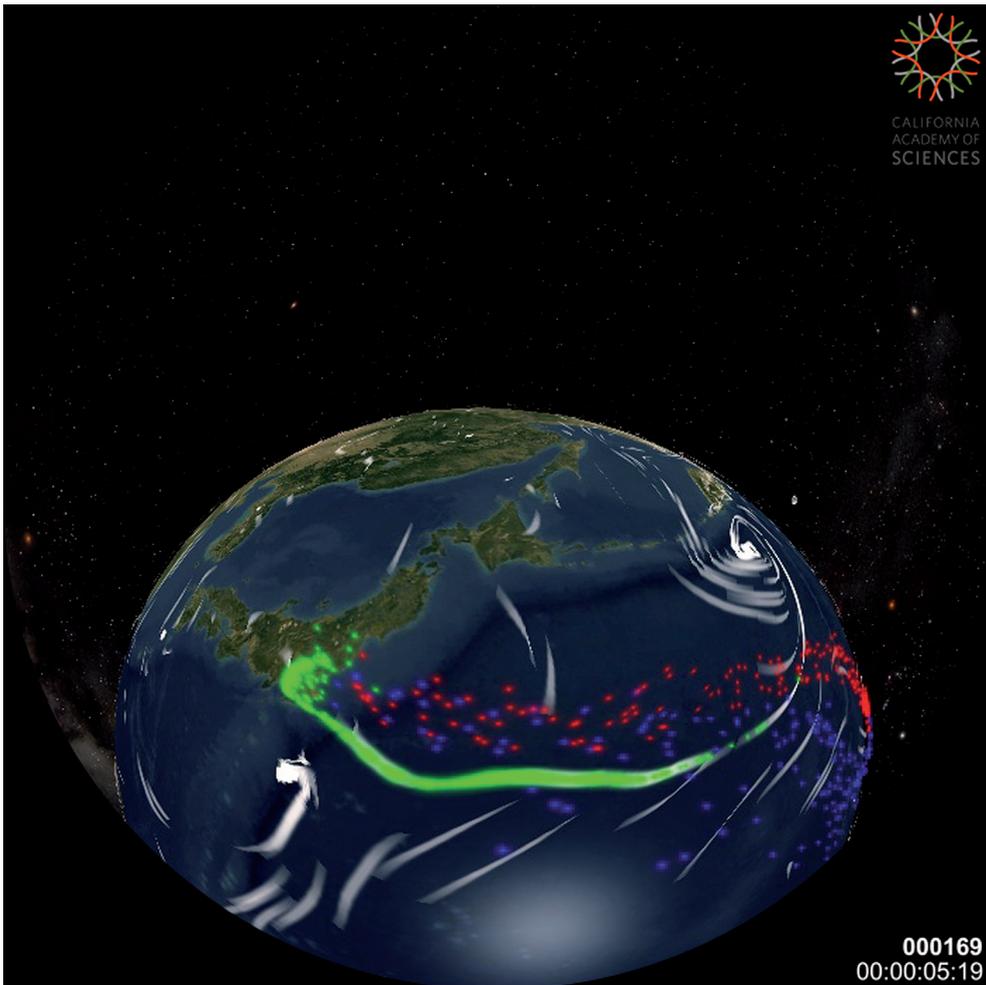
Die Objekte wie Sterne, Planeten, Umlaufbahnen u.ä. sind in der Programmoberfläche gut erreichbar und nutzbar. Bei den größeren Anbietern von Planetariumssystemen ist sogar der gesamte «Digital Universe Catalogue» (AMNH 2011) umgesetzt.

Zusätzlich dazu können bei einigen Systemen Simulationen angewendet werden, die in vereinfachten Modellen die Kollision von Galaxien errechnen können, oder virtuelle Partikel in der Erdatmosphäre je nach aktueller Wetterlage (die aus dem Internet heruntergeladen werden kann) verteilen. Auf diese Weise können aktuelle Geschehnisse für das Publikum veranschaulicht werden (siehe Abb. 1). In der Regel lassen sich auch mit einem Echtzeitsystem selbstablaufende Shows produzieren, basierend auf

1 *Frames per second (fps)*: Bilder pro Sekunde.

2 *Generator locking device*: Zur Synchronisation analoger Videoquellen durch einen externen Taktgeber.

3 Vom Grafikkartenhersteller NVidia eingeführte Technik zur Synchronisation der Anzeigepixelabtastung mit einer Synchronisationsquelle; unter: [http://www.nvidia.de/object/feature\\_frame\\_lock\\_de.html](http://www.nvidia.de/object/feature_frame_lock_de.html) [25.07.2011].



1 Partikel-Simulation in der Erdatmosphäre nach dem Fukushima-Unfall: Grüne Partikel zeigen die Verteilung in niedrigen Atmosphärenschichten, rot und blau in höheren Schichten. Die Simulation erfolgte unter Einbeziehung aktueller Wetterdaten in Echtzeit auf einem *DigitalSky*-System mit *FlowViz*.<sup>4</sup>

einer einfachen Skriptsprache oder einer Timeline, die beschreibt, wann etwas passiert.

Solche zeitbasierte Information kann genutzt werden, um eine virtuelle Welt in das System einzuladen. Die Bildgeneratorsoftware kennt also die 3-D-Welt und die Position aller Objekte in ihr. Weiterhin weiß das System, wo die Projektoren stehen und welchen Teil der Kuppel jeder Projektor bespielt. Die Bildgeneratorsoftware berechnet durch Verteilung

der Rechenlast auf die einzelnen Zuspieldreher eine 3-D-Welt so, wie sie aussähe, wenn man sie auf die Kuppel projiziert. Dabei ist die virtuelle Kamera (mit einer Fisheye-Linse als Abbildungsvorschrift) in der Kuppelmitte.

Die Objekte der 3-D-Welt werden (im Falle der astronomischen Objekte) entweder vom Hersteller mitgeliefert oder sind mit einem 3-D-Programm erzeugbar, so dass auch leicht eigene Objekte eingebunden werden können. Durch die Position der virtuellen Kamera in der Kuppelmitte ist dort auch der Ort, an dem die Projektion verzerrungsfrei erscheint. Aus allen anderen Orten in der Kuppel schaut man

<sup>4</sup> Vielen Dank an Tim Florian Horn von der California Academy of Sciences, San Francisco, dafür, dass er dieses Bild zur Verfügung gestellt hat.

schräg oder mit unterschiedlichen Entfernungen auf die Projektionsfläche, was zu Verzerrungen führen kann. Ist dieser Effekt nicht zu stark, ist die menschliche Wahrnehmung in der Lage, ihn zu ignorieren.

Trotz der Inkompatibilitäten in der Realisierung der Echtzeitsysteme sind die technischen Rahmenbedingungen erstaunlicherweise in allen Kuppeln sehr ähnlich: Es gibt multiple Projektoren, die je nach Typ und Anzahl auf verschiedene Weisen aufgestellt sein können. Man spricht von der Projektorkonfiguration einer Kuppel. Die Gemeinsamkeit aller Konfigurationen ist die Tatsache, dass alle Projektoren zusammen die gesamte Kuppel mit Bild bespielen. Jeder Projektor wird seinerseits von Grafikrechnern bespielt, auf denen die Bildgeneratorsoftware aktiv ist. Die Software hat in diesem Verbund mehrere Aufgaben: Zum einen müssen die Bildteile passend zur Projektorkonfiguration berechnet und die Rechner synchronisiert werden, zum anderen braucht der Nutzer eine Oberfläche zur Steuerung und Befehls-eingabe. Zudem müssen die für Interaktion nötigen Controller abgefragt werden. Der in Bezug auf Kompatibilität limitierende Faktor ist also die Mächtigkeit der Bildgeneratorsoftware.

## Möglichkeiten der Interaktion

Im Gegensatz zu einem Netzwerk-basierten Spiel, in dem eine hohe Anzahl von Nutzern an verschiedenen Rechnern gleichzeitig interagieren kann, gibt es für Interaktion in einer Kuppel nur einen einzigen Bildschirm, nämlich die Kuppel selbst. Hat ein kleines Kuppeltheater wie der Mediendom in Kiel mit 9 Metern Durchmesser schon 64 Sitzplätze, muss sich ein Entwickler interaktiver Anwendungen viele Gedanken um die Realisierbarkeit einer Multi-User-Anwendung machen. Da die Kuppel als einzige Ausgabefläche für alle Spieler ein limitierender Faktor sein kann, beschränken sich im Moment interaktive Anwendungen (wie auch das Spiel in der Show ALIEN ACTION) auf einzelne Benutzer, die mit Standard-Controllern wie z. B. einem Joystick agieren.

Dieser Umstand bewirkt eine Trennung desjenigen, der die Kuppel interaktiv bedient, vom Rest der Zuschauer. Diese können nur noch durch Anfeuern oder vergleichbare Tätigkeiten (also rezeptiv) auf die Interaktion einwirken, die direkte Kontrolle haben sie allerdings nicht mehr. Die Situation ist also vergleichbar mit einem Sportereignis, bei dem die Zuschauer auf den Rängen die Mannschaft zwar anfeuern, aber nicht mitspielen können. Und obwohl die meisten Zuschauer nicht direkt die Interaktion

erleben können, muss für alle die Visualisierung unterbrochen werden, damit die eine Person den Controller erhalten und nutzen kann.

Der besondere Reiz von Interaktion in Kuppeln ist aber das Einbinden vieler Zuschauer gleichzeitig, was die Trennung der Interagierenden und passiv Erlebenden aufhebt.

Eine bereits existierende technische Möglichkeit zur Realisierung dieser Aufgabe stellen Knöpfe in den Armlehnen der Sitze dar, mit denen jeder Zuschauer Einfluss auf das Kuppelgeschehen nehmen könnte: Quiz-Shows oder Abstimmungen sind erste Ideen für eine solche Multi-User-Anwendung.

Aufgrund eines Mangels an Kuppeltheatern mit solchen Einrichtungen und ihren Anwendungen stellt sich die Frage, welche weiteren Möglichkeiten sich bieten würden:

Durch die Bildung von Mannschaften könnten Objekte durch mehrere Personen gleichzeitig gesteuert werden, was wiederum komplexe Abstimmungsprozesse der Beteiligten untereinander erfordern würde, aber auch große Erfolgserlebnisse verspricht im Sinne einer gemeinsamen Produktion (in der kommunikationswissenschaftlichen Bedeutung dieses Wortes). Wenn es gelänge, jedem Zuschauer einen individuell steuerbaren Avatar zu geben, kann von Interaktion im engeren Sinne gesprochen werden. Dieses lässt sich aber bei großen Zuschau-erzählen allein aufgrund der Anzahl der Zuschauer nur schwierig realisieren (große Fulldome-Kuppeln haben über 250 Sitzplätze). Ein erster Versuch wurde bereits unternommen (s. u.).

Grundsätzlich hat die Art und Weise der Interaktion mit drei Dingen zu tun: dem inhaltlichen Thema der interaktiven Anwendung (das hier nicht diskutiert werden soll), den zur Verfügung stehenden Controllern und der softwaretechnischen Umsetzung ihrer Funktionalität. In technischer Sicht ist man also auch hier wieder vom Funktionsumfang der Bildgeneratorsoftware beschränkt. In der Regel sind vom Hersteller einige Controller in ihrer Grundfunktionalität im System verankert, so bieten viele die Nutzung von Joysticks oder Playstation-Controllern. Auch können spezielle Bedienelemente wie Schalt-pulte (bei Hybrid-Systemen mit opto-mechanischem Projektor) oder frei belegbare Buttons, Dreh- und Schieberegler Verwendung finden.

## Erweiterung der vorhandenen Möglichkeiten: MVC-Pattern und Ausnutzung von *Warping*

Zu einer interaktiven Anwendung gehört aber noch mehr: Normalerweise gibt es eine komplexe Regelmenge, in der sich der gesamte logische Szenenaufbau befindet. Dies kann eine Beschreibung der 3-D-Welt in Bezug auf physikalische Gesetze wie z. B. Gravitation sein: durch die Regelmenge haben Objekte auf einmal ein Gewicht und verhalten sich entsprechend; auch könnte es passieren, dass Objekte kollidieren, dann zerbrechen oder aneinander haften bleiben. Der Szenenaufbau ist also deutlich mehr als nur eine Sammlung von Positionen von Objekten, und es kann nötig sein, die Szene so zu beschreiben, dass Simulationen stattfinden können und Objekte automatisiert interagieren. Die Regelmenge steuert also die gesamte Welt und definiert insbesondere, was nicht passieren darf oder kann.

Es erscheint also sinnvoll, die Funktionalität eines Fulldome-Bildgenerators zu erweitern, denn die meisten Systeme sind auf die Darstellung von astronomischen oder terrestrischen Daten spezialisiert.

Die Erweiterung kann natürlich durch das Ersetzen des gesamten Bildgenerators durch eine Spiele-Engine<sup>5</sup> erfolgen. Es ergibt sich dabei aber ein großes Problem: das Bild soll auf eine Kuppel projiziert werden. Die Hersteller von Fulldomesystemen haben viele Jahre investiert, um den *Warping*<sup>6</sup>-Algorithmus und das *Edge-Blending*<sup>7</sup> für viele Grafikkrener und die zugehörigen Projektoren möglichst flexibel und gut konfigurierbar zu machen. Die Annahme, dies schnell eigenständig erstellen zu können, scheint vermessend.

<sup>5</sup> Bekannte Spiele-Engines sind die «Unreal»-Engine, für den Spieleprogrammierer als SDK verfügbar unter <http://www.udk.com> (*Unreal Development Kit*). Auch die aus dem VR-Bereich stammende Engine «Unity» (<http://www.unity3d.com>) wäre für den Kuppelbereich geeignet, da sie verteilte Bildberechnung unterstützt [25.07.2011].

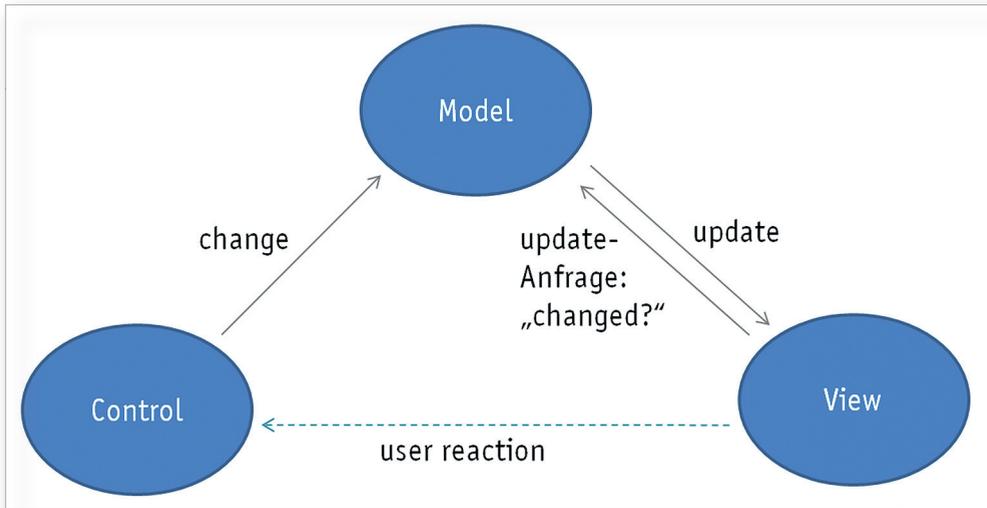
<sup>6</sup> Das Erzeugen von verzerrungsfreier Bildwiedergabe an einer Kuppel. Dies erfordert ein Verzerren des Bildes im Grafikkrener und eine genaue Justierung des Projektors, um ein Stück Kuppel so zu bespielen, dass das Bild für den Betrachter verzerrungsfrei erscheint.

<sup>7</sup> Bei mehreren Projektoren, die ein übergangsfreies Gesamtbild erzeugen sollen, müssen die Ränder jedes projizierten Teilbildes mit einer weichen Maske versehen werden. Diese Übergangsbereiche sind die Überlappungsbereiche der Projektoren und müssen justierbar sein.

Die einzige Möglichkeit, bei Nutzung einer Spiele-Engine dieses Problem zu umgehen, ist die Nutzung von einem Projektor in der Kuppelmitte, der mit einem Fisheye projiziert. Auf diese Weise braucht man kein *Warping*, allerdings ist die Auflösung und Helligkeit des Kuppelbildes auf die des einen Projektors beschränkt. Für große Kuppeln ist dies nicht praktikabel.

Ein vielversprechendes Ziel ist das Erweitern des vorhandenen Kuppelsystems: hier ist das *Warping* bereits komfortabel gelöst. Die Erweiterung eines Systems setzt eine Programmierschnittstelle (API) voraus.

- Die Idee der Erweiterung ist die des Programmierschemas «Model-View-Control» (auch MVC-Designpattern; vgl. Gamma et. al. 1994: 4-6). Dieser Begriff der Informatik beschreibt eine Vorgehensweise bei der Erstellung grafischer Anwendungen. Die Programmierung teilt sich dabei in drei Teile (siehe Abb. 2): Das Model enthält die Eigenschaften sämtlicher Objekte (wie z. B. Position, Geschwindigkeit und weitere Zustände) und die gesamte Logik und die Regeln der Welt, die simuliert werden soll (s.o.).
- Control nimmt die Benutzereingaben über Eingabegeräte (*Controller*) entgegen. Es interpretiert diese soweit, dass das Model mit den Informationen arbeiten kann und schickt dann diese Eingaben an das Model. Dieses reagiert im Sinne der Regelmenge, die im Model spezifiziert ist, d. h. entscheidet, ob die Eingaben erlaubt sind und verändert die Welt durch passende Aktionen.
- Der View fragt das Model in regelmäßigen Abständen, ob sich etwas in der Welt geändert hat. Ist dies der Fall, schickt das Model ein «update», was den View veranlasst, seine Ausgabe zu aktualisieren, z. B. durch Neuzeichnen der grafischen Ausgabe. Es können beliebig viele Views beim Model angemeldet sein. Dies ist dann sinnvoll, wenn verschiedene Sichten auf ein Modell dargestellt werden sollen. Beispiele wären bei einer simulierten Welt zum einen die grafische Ausgabe, zum anderen ist aber auch ein akustischer View denkbar: in dem Fall wird der Ton, der sich an 3-D-Objekte anheften lässt, in dem Surround-Audiosystem der Kuppel an der richtigen Stelle ausgegeben. Ein weiterer View könnte eine textbasierte Ausgabe auf einem Kontroll-



2 Model-View-Control Designpattern.

monitor sein. Der Benutzer reagiert auf die Veränderung der Views durch Eingaben mit den Controllern.

Wie nützt dieses Schema nun der Erstellung interaktiver Anwendungen in Kuppeln? Grundsätzlich wird das so erstellte Programm die Kontrolle über die Bildgeneratorsoftware erhalten. Control kann, da es selbst entwickelt ist, mit den Controllern umgehen, für deren Steuerung es implementiert wurde, d.h. die Einbindung von bislang nicht ansprechbaren Controllern ist möglich. Das Model reagiert entsprechend der in ihm beschriebenen Welt und leitet die Objektpositionen, die aus der Regelmenge resultieren, an den View weiter: der View ist dann die Bildgeneratorsoftware der Kuppel, die ohnehin vorhanden ist. In dieser Software werden dann (wie bei jeder anderen Echtzeitanwendung auch) Objekte geladen, die schließlich gemäß ihrer (vom Model gesteuerten) Position ausgegeben werden. Da der View aber bereits über das Waring und Edge Blending verfügt, muss der Programmierer sich um den gesamten Aufgabenbereich der Kuppeldarstellung nicht kümmern. Diese Vorgehensweise kann sehr effizient sein, ist aber auch wieder durch die Offenheit und Mächtigkeit der API-Schnittstelle des Kuppel-Bildgenerators beschränkt.

Weiterhin erlaubt dieses Verfahren eine effiziente Aufteilung der Aufgaben bei der Erstellung einer

interaktiven Anwendung: gemäß der MVC-Struktur werden alle Dinge, die im View dargestellt werden (der Content), von 3-D-Grafikern als Echtzeit-Modelle erzeugt. Die Programmierung findet getrennt davon statt, es muss nur eine Schnittstelle definiert werden, so dass beide Produktionsteile hinterher zusammengefügt werden können.

### Verschiedene Typen von Echtzeit

In einem ersten Fazit soll zusammengefasst werden, dass es verschiedene Typen von Echtzeitvisualisierung gibt. Gemeinsam ist allen, dass in einer Welt Objekte gesteuert und dann dargestellt werden. Die Art und Weise der Erstellung der Objekte ist für alle gleich.

Allerdings kann die Steuerung auf verschiedene Weisen passieren:

- mit einer Skriptsprache oder Timeline,
- mit Controllern interaktiv oder
- mit einer externen Anwendung, die ein komplexes Regelsystem ermöglicht und die den Bildgenerator erweitert.

Das bedeutet insbesondere, dass Interaktion nicht zwingend Teil von Echtzeitvisualisierung ist. Zudem gibt es meistens selbstablaufende und interaktive Szenen im Wechsel, so dass grundsätzlich von der Nutzung von vorbereitetem Video nicht abgeraten werden soll.

## Immersion von Spielen im Kuppelmedium und weitere Möglichkeiten

Der hermetische Ausschluss der Außenwelt und die vollständige Umgebung der Nutzer durch die Form des Mediums können zu einer guten räumlichen Illusion führen, die der Beschreibung von Grau bereits sehr nahe kommt:

«Dazu gehört neben den Mitteln des Illusionismus insbesondere das den Gesichtswinkel möglichst vollständig ausfüllende Bildformat und die Ansprache möglichst vieler anderer Sinne. Die hierdurch mögliche Suggestion, die den Betrachter gewissermaßen in den Bildraum eintauchen lässt, vermag die Subjekt-Objekt-Beziehung für einen gewissen Zeitraum aus den Angeln zu heben und dem «Als-Ob» im Bewusstsein Konsequenz zu verschaffen.»  
(2002: 23f.)

Die Illusion als eine der Bedingungen für Immersion wird durch den Aufbau der Kuppel also unterstützt. Weitere Bedingungen sind sicherlich durch den Inhalt bestimmt: Hier unterscheiden sich Spiele oder andere interaktive Anwendungen grundlegend von den bislang vorhandenen nicht-interaktiven (und somit in Bezug auf technische Interaktion passiv konsumierten) Inhalten. Die eher spärlichen Untersuchungen zu Wirkung von Inhalten in der Kuppel befassen sich aber nur mit den nicht-interaktiven Inhalten, so dass untersucht werden sollte, inwieweit die Interaktion eine neue Form der immersiven Erfahrung in die Kuppeln bringt bzw. ob der Nutzen so hoch ist, dass dies ein erfolgversprechendes Konzept für die Zukunft des Mediums sein kann. Diese generelle Fragestellung ist aber nur schwer mit den kaum vorhandenen Spiele-Prototypen zu beantworten.

Hier treffen nicht nur die Interessen der Techniker und Entwickler von Kuppeln und Bildgeneratoren aufeinander, sondern auch die der Zuschauer. Die Akzeptanz von Echtzeitanwendungen beim Publikum ist ein Thema, das mit Erwartungshaltung zu tun hat. Vom Kino kennen die Zuschauer ein lineares Medium in hoher Qualität, das passiv genutzt werden kann. Die Zuschauer können sich zurücklehnen und konsumieren. Bei Visualisierungen jeder Art in Kuppeln ist dies anders: Vom Zuschauer wird eine aktive Bereitschaft verlangt,

die Kuppelfläche wahrnehmen zu wollen, sich also aktiv umzuschauen; das aus dem Kino bekannte Verhalten des Zurücklehns scheint in einem Medium, das den Zuschauer komplett umgibt, nicht angebracht.

Erschwerend kommt hinzu, dass die visuellen Möglichkeiten eines Echtzeitsystems häufig hinter den der vorberechneten Darstellungen zurückfallen; die technischen und visuellen Fortschritte der Computerspiele der letzten Zeit werden erst viele Jahre später in Fulldome-Systemen umgesetzt und dienen dann der Darstellung von astronomischen Inhalten<sup>8</sup>. Die Flexibilität des Systems beim Erstellen von Spieleanwendungen beliebiger Themen steht nicht im Fokus der Entwickler dieser Systeme. Außerdem wird der lineare Erzählfluss unterbrochen, wenn vor der Interaktion die Nutzung der Controller erklärt wird. Wenn (wie vor dem Spiel aus ALIEN ACTION) ein Joystick an einen Zuschauer vergeben und erklärt wird, muss dafür etwas Licht in der Kuppel angeschaltet werden, um den Joystick sichtbar zu machen. Dies führt auch zu einem Bruch der Immersion. Diese Unterbrechung muss folglich einen Mehrwert haben, der vom Publikum häufig nur schwer erkannt wird. Ein paar Fragen sollen zu diesem Themenkomplex gestellt werden:

- Wie erzeugt man mit geschicktem Game-Design Multi-User-Anwendungen oder Spiele auf dem einen großen Display? Ein erster Versuch wurde im Planetarium Hamburg 2010 unternommen, als die über 200 Zuschauer in der Kuppel mit ihren iPhones jeweils einen Avatar steuern konnten (Meyer, 2010: k.S.). Bei einer so hohen Zahl an Zuschauern stellt sich aber die Frage, ob diese große Menge an *Response* der eigenen Aktionen und der der anderen Spieler noch in dem Maße wahrnehmbar ist, dass die Interaktion einen Mehrwert hat. Eine Herangehensweise zur Lösung wäre eine stärkere Lenkung der Spieler, die aber zu Recht die Frage nach Sinn bzw. Wichtigkeit der Interaktion im Kontext der Show aufwirft.
- Wie funktioniert die immersive Rundum-Wahrnehmung bei der gleichzeitigen Steuerung mit einem Controller? Wie geht der Zuschauer, der an das Medium nicht gewöhnt ist, mit dieser Anforderung um? Die Nutzung des Joysticks beim Spiel von ALIEN ACTION genauso wie jede andere Form von Interaktion zeigt in verschiedenen Planetarien die Tendenz, dass die Zuschauer damit

<sup>8</sup> Vgl. die Einbindung einer Landschafts-Engine (*ProLand*) im «SkyExplorer 3» der französischen Firma RSA Cosmos, unter: <http://www.rsacosmos.com/imagesus/sky-explorer/sky-explorer.pdf> [25.07.2011].

überfordert sind<sup>9</sup>. Spiele-Designer sind also auch hier gefordert, möglichst einfache Aufgaben mit möglichst einfachen Controllern zu entwickeln, so dass diese für ein breites Publikum keine Hürde darstellen. Die Probleme sind aber nicht nur technischer Natur (in Bezug auf die Bedienung des Controllers), sondern auch psychischer: Die Angst, sich vor einem Publikum zu blamieren, führt dazu, dass sich viele nicht freiwillig für die Nutzung des Joysticks melden. Umso wichtiger scheint es, dass bald mehr Multi-User-Anwendungen entwickelt werden.

- Wie gehen die Zuschauer im Falle von sehr vielen Mitspielern miteinander um? Diese Fragestellung ist zum einen die Frage nach sozialem Verhalten in Gruppen, aber auch eine Frage nach dem Verhalten, wenn nicht die gesamte Szene gleichzeitig im Blick gehalten werden kann. Eine genauere Spezifikation der Frage scheint im Moment nicht möglich.
- Ist die Nutzung eines Controllers zur Steuerung von Echtzeit-Inhalten wirklich immersiv? Immersion ist insb. das Zusammenwirken von vielen Teilaspekten: neben der räumlichen (visuellen) Illusion müssen auch alle anderen Sinne entweder in der Logik der virtuellen Szenerie bedient oder möglichst gut ausgeblendet bzw. nicht genutzt werden. Die ständige haptische Erfahrung eines Controllers kann in der Totalitätserfahrung der Kuppel eher störend wirken. Mit Spannung werden hier die ersten gesteuerten Kuppelzuspeler gesehen (siehe <http://www.youtube.com/watch?v=T1yxfeMKd6I> [25.07.2011]), die allerdings noch nicht deutlich aufzeigen, welche Möglichkeiten man in Zukunft erschließen kann und ob diese Form der Anwendung praktikabel ist.

## Perspektiven von Echtzeitproduktion und Spielen in Kuppeln

Die Frage nach der Zukunftsträchtigkeit von Echtzeitanwendungen ist auch immer eine Frage nach der Zukunft der Projektionskuppeln an sich. Diese sind als immersives Medium spätestens seit dem Gottorfer Globus (vgl. Lühning 1997), mindestens aber seit der Realisierung eines künstlichen Sternenhimmels in einem ersten Planetarium (1923 in Jena; vgl. Deutsches Museum 2011) etabliert (beide Installationen waren interaktiv durch den Vorfüh-

rer bedienbar), dennoch finden sie bei Grau keine Erwähnung. Die Perspektiven von Echtzeitanwendungen in Kuppeln sind von hochgradig fachübergreifender Natur. Im Folgenden sollen einige besonders stark beeinflussende Größen genannt werden, die für zukünftige Entwicklungen entscheidend sein können:

- Für mehr Echtzeitanwendung in Kuppeln spricht die Tendenz der grundsätzlich steigenden Pixel-Auflösung von digitaler Kuppelprojektion. So wurde 2010 die erste Kuppel in Europa, die fast 8000x8000 Pixel (8k, also 64 Megapixel) Auflösung erreicht, in Münster eröffnet. Auch ist der Kuppeldurchmesser z.T. sehr groß, waren es 2010 bei der Eröffnung des Planetariums am Ehime Science Center in Japan noch 30 Meter Durchmesser, die zum Titel weltgrößtes Planetarium führten, wurde dieser Wert 2011 durch die Kuppel des Nagoya Planetariums (auch Japan) mit 35 Metern abgelöst. Bei solchen Kuppeln ist nicht nur die Auflösung, sondern auch die Lichtleistung der Projektoren eine entscheidende Größe. Die Auflösung von ca. 8k auf der gesamten Projektionsfläche erfordert viele lichtstarke und hochauflösende Projektoren, und sorgt so für ein scharfes lichtstarkes Bild. Da die Schärfe aber insb. durch die hohe Auflösung erzielt wird, benötigt man aber auch eine große Anzahl von Zuspelerrechnern, um die Fläche auch in Echtzeit bespielen zu können. Vorerendertes Video braucht in der Produktion schon vielmals so lange wie bei der heute noch üblichen 4k-Auflösung (16 Megapixel) und fordert entsprechend mehr Speicher, um die Einzelbilder zu lagern. Echtzeitproduktion ist hier eine mögliche Lösung, denn die Eigenschaft von 3-D-Modellen, als 3-D-Vektordaten vorzuliegen, erlaubt anders als beim Hochskalieren des gerasterten Bildes des digitalen Films ein verlustfreies großes Darstellen. Einige Dinge sind aber zu beachten: die Auflösung der Texturen der 3-D-Modelle muss entsprechend höher sein, denn Texturen sind in der Regel (Pixel-)Bildinhalte. Auch die Geomet-

<sup>9</sup> Dies ist das Ergebnis von nicht-repräsentativen stichprobenartigen Gesprächen mit Planetariumsmitarbeitern aus dem deutschsprachigen Raum, die ich 2011 auf der Arbeitstagung deutschsprachiger Planetarien in Wien auf das Thema ansprach.

rie (die polygonale Struktur) der Objekte kann bei einer so hohen Auflösung die Notwendigkeit einer Detailerhöhung mit sich bringen.

- Die Akzeptanz, dass Echtzeitanwendungen für den Betrieb eines digitalen Planetariums notwendig sind, ist inzwischen bei allen Herstellern sehr groß. So bieten alle eine Vielzahl an Möglichkeiten der Echtzeitvisualisierung, Simulation oder Interaktion mit verschiedenen Controllern. Diese eigentlich hervorragenden Voraussetzungen werden durch die Tatsache beinahe annulliert, dass der Markt der digitalen Projektionssysteme sehr klein und umkämpft ist. Jeder Hersteller hat eine eigene Herangehensweise für die Produktion und somit eine bestimmte Menge an den oben genannten Möglichkeiten (die sogenannte *feature list*). Direkt daraus resultiert eine Inkompatibilität zwischen den Systemen, die nicht nur die prinzipielle Umsetzung von Inhalten, sondern auch die Möglichkeiten betrifft, die überhaupt genutzt werden können. Eine Echtzeitanwendung, die auf allen Systemen laufen soll, kann also maximal die kleinste gemeinsame Menge an Funktionalitäten nutzen. Da es weiterhin keine Standards für Echtzeit-Anwendungen in Kuppeln gibt, muss ein Produzent für alle Systeme entwickeln, was aus Kostengründen nicht möglich ist bzw. auch von den Herstellern der Bildgeneratoren nicht gewünscht scheint: Anfragen von verschiedenen Seiten (professionelle Produktionsstudios genauso wie Planetariumstechniker) wurden entweder gar nicht oder nur mit äußerster Zurückhaltung beantwortet; erklärbar ist dies nur durch massiven Konkurrenzdruck auf dem Markt. Entwickelt ein Produzent nur für ein System, schränkt sich die Zahl der möglichen Kunden zu sehr ein. Also wird der einfache Weg gewählt und Fulldome-Video produziert (dies ist aufgrund vorhandener Kompatibilität auf allen Systemen einfach abspielbar). Diese Umstände führen dazu, dass

<sup>10</sup> Siehe die NVidia-Webseite über Tessellation unter [http://www.nvidia.de/object/tessellation\\_de.html](http://www.nvidia.de/object/tessellation_de.html) [26.7.2011].

<sup>11</sup> Ein gutes Beispiel für Echtzeit-Landschaften ist das 2012 erscheinende Computerspiel FAR CRY 3 (Ubisoft, F 2012), das in seinen Möglichkeiten weit vor dem liegt, was Fulldome-Systeme heutzutage leisten. Eine Videovorschau zu diesem Spiel ist zu finden unter <http://far-cry.ubi.com/far-cry-3/de-DE/> [26.7.2011].

Echtzeit fast völlig aus dem Produktionsprozess von professionellen Fulldome-Produzenten verschwunden ist. Die Einbindung von Echtzeit in den Produktionsprozess hat sogar noch weitere Auswirkungen: zum einen wäre eine weitere Produktionsparte einzubinden, nämlich Programmierer, die interaktive Anwendungen erstellen können. Für die Erstellung von Kuppel-Film sind sie natürlich nicht notwendig. Zum anderen würden die gewohnten Pfade verlassen werden müssen, da Echtzeit-3-D-Modellierung notwendig wird, die sich in bestimmten Aspekten vom Modellieren für 3-D-Filme unterscheidet: durch die Beschränkung der zur Verfügung stehenden Renderzeit pro Frame (nämlich gemäß der Framerate des Echtzeit-Systems) dürfen Modelle einen gewissen Komplexitätsgrad nicht überschreiten. Dies betrifft sowohl die polygonale Struktur (Geometrie), als auch Textur, Reaktion der Oberfläche auf Licht (*Shading*), Interaktion mit anderen Objekten, z.B. Kollision, aber auch Licht und Schatten, Reflektionen usw. Um den Komplexitätsgrad niedrig zu halten, gibt es vielfältige Optimierungsmöglichkeiten, z.B. bei festen Beleuchtungszuständen das «Backen» (engl. *baking*) von Texturen: Sämtliche Licht-/Schattenverhältnisse, teilweise sogar Reflektionen werden als normale Farbinformationen in Oberflächen integriert, so dass sie vom Bildgenerator nicht mehr berechnet werden müssen. Da Optimierungen aber stets mit Qualitätsverlust einhergehen, ist hier komplexes Know-How gefragt, um Echtzeit-Modelle möglichst visuell ansprechend zu erzeugen. Hilfe könnte die Technik der prozeduralen Modellierung bringen, die für Landschaftsmodelle wichtig ist, oder die Tessellation-Technik, die durch den DirectX11-Standard verfügbar wird<sup>10,11</sup>. Diese Techniken finden erst selten im Fulldome-Bereich Anwendung, es gibt bislang nur ein System, das diese nutzt (s. Fußnote 8). Mit diesen Techniken unterscheidet sich die Produktion solcher Modelle und Szenen massiv von vorgerendertem Film: Dort müssen Frames nicht innerhalb weniger Millisekunden berechnet sein, sondern können viele Minuten (teilweise sogar Stunden) rechnen, was komplexe Modelle, volumetrische Objekte oder komplexe Beleuchtungsszenarien erlaubt. Alle Berechnungen können stattfinden und müs-

sen nicht optimiert werden. Gerade im Bereich möglichst natürlich wirkender Darstellung von z. B. Landschaften wird daher bevorzugt vorgezogen.

- Ein weiterer Punkt kommt erschwerend hinzu: Die einzigen Spielstätten für digitale Kuppelprojektion sind (noch) Planetarien. Die Hersteller von Kuppel-Bildgeneratoren haben häufig nicht den Fokus ihrer Entwicklung auf den Möglichkeiten, die eine echtzeitfähige Kuppel als eine VR-Umgebung bietet, sondern auf der möglichst einfachen Darstellung von astronomischen Inhalten. Dies ergibt sich u. a. aus der Erwartung, die das Publikum an ein Planetarium hat. Diese ist nicht nur medial, sondern auch inhaltlich geprägt: man will etwas über den Sternenhimmel erfahren. Auch hier treffen wieder zwei Welten aufeinander: Spiele oder interaktive Wissensvermittlung haben nicht zwingend astronomische Themen. Es wäre wünschenswert, dass viele Planetarien für andere Themen offen sind und diese Themenvielfalt auch bewerben; der Trend geht langsam in diese Richtung. Trotzdem scheinen die astronomischen Features von Bildgeneratoren immer noch im Fokus der Entwickler zu liegen. Damit stehen die Hersteller qualitativ zwangsläufig hinter den aktuellen Entwicklungen der Spiele-Engines zurück (s.o.) und sind damit für Entwicklungen von virtuellen Szenen oder Spielen nichtastronomischen Inhaltes unattraktiver als die Spiele-Engines.

## Fazit

Es gibt eine ausgesprochen große Vielfalt an Möglichkeiten für digitale Projektionskuppeln Echtzeitvisualisierungen zu erstellen. Auch wenn gerade für Produzenten mit geringen Informatikkenntnissen die technischen Hürden groß erscheinen, ist es umso wichtiger, die Produzenten und auch am Kuppelformat interessierte Entwickler «normaler» PC-Spiele gut auszubilden und sie mit dem Medium vertraut zu machen. Dies hat auch ein Weiterdenken in Bildgestaltung und filmischer Montagetheorie zur Folge, die in diesem Artikel nicht diskutiert worden sind, die aber seit einigen Jahren in Kiel erforscht werden (vgl. Hertling 2007: 8-10; Buczek 2007: 29-32). Die interaktiven Anwendungen in Kuppeln werden bei häufiger Nutzung helfen, das Medium nicht nur den Produzenten, sondern auch dem Publikum vertrauter zu machen und die Alleinstellungsmerkmale

von Kuppeln hervorheben: Kuppeln sind kein Kino, sondern können bei geeigneter Nutzung viel mehr bieten.

Wichtig für die Akzeptanz sind aber neben der visuell ansprechenden Umsetzung von Echtzeitszenen auch die Möglichkeiten der Echtzeitsysteme selbst: ohne einen Standard wird sich die Nutzung von Interaktion in Kuppeln immer auf wenige Prototypen beschränken. Dort ist also noch viel zu tun. Ein weiterer Lösungsansatz wäre eine offene Engine, die neben Kuppelwarping und -edging bei beliebiger Projektorkonfiguration auch verteiltes Rendern komplexer Szenen und die Synchronisation der beteiligten Computer unterstützt. Von so einem Ansatz ist die Fulldome-Welt im Moment aber noch weit entfernt.

Ändert sich der Mangel an Standard oder Offenheit nicht, werden die Produzenten eigene Wege gehen und sich von den Kuppelsystemen lösen: So ist das Spiel in der Mediendom-Produktion ORCHIDEEN – WUNDER DER EVOLUTION (D 2009) für einen einzelnen Bildschirm / Beamer konzipiert, damit es möglichst häufig verkauft werden kann. Dieser Weg mag aus Marketinggesichtspunkten sinnvoll erscheinen, führt aber die Entwicklung weg vom Kuppelsystem zurück zur Einzelnutzer-Anwendung am Bildschirm. Dies kann nicht das Ziel der Hersteller der Bildgeneratoren und der Fulldome-Produzenten sein.

## Literatur

- Akenine-Möller, Tomas/Haines, Eric/Hoffman, Naty (2008) *Real-time Rendering*. Wellesley: Peters.
- AMNH (2011) *American Museum of Natural History, The Digital Universe Catalogue*. Online: <http://www.haydenplanetarium.org/universe> [24.07.2011].
- Buczek, Isabella (2007) Findung einer visuellen Sprache. In: *Jahrbuch immersiver Medien 2007*. Hg. von Eduard Thomas. S. 29-32
- Buttazzo, Giorgio C. (2004) *Hard Real-time Computing Systems: Predictable Scheduling Algorithms and Applications (Real-Time Systems)*. Berlin: Springer.
- Deutsches Museum (2011) *Das Wunder von Jena*. Online: <http://www.deutsches-museum.de/ausstellungen/naturwissenschaft/astronomie/planetarium/geschichte-des-planetarium/> [25.07.2011].
- Gamma, Erich et. al. (1994) *Design Patterns*. Amsterdam: Addison-Wesley.
- Heinsohn, Ralph & Bünning, Dominic (2007) Alien Action. In: *Jahrbuch immersiver Medien 2007*. Hg. von Eduard Thomas. S. 54-58.

- Hertling, Peter (2007) Auf der Suche nach dem 360°-Paradigma. In: *Jahrbuch immersiver Medien 2007*. Hg. von Eduard Thomas. S. 8-10
- Grau, Oliver (2002) *Virtuelle Kunst in Geschichte und Gegenwart*. Berlin: Reimer.
- Kjär, Heidi & Rienow, Jürgen (2009) Fulldome-Visualisierung und Immersion in der Lehre. In: *Jahrbuch immersiver Medien 2008/09*. Hg. von Eduard Thomas. S. 60-67.
- Lühning, Felix (1997) *Gottorf im Glanz des Barock, Teil IV: Der Gottorfer Globus und das Globushaus im ,Newen Werck*. Schleswig: Eigenverlag Schleswig-Holsteinisches Landesmuseum.
- Meyer, Axel (2010) *360touch.it*. Online: <http://360touch.it> [25.07.2011].
- Rienow, Jürgen (2002) *Skalierbare Synchronisation eines Rechner-Clusters über IP-Netzwerke*. Studienarbeit. Christian-Albrechts-Universität.
- Wikipedia (2011a) *Echtzeit*. Online: <http://de.wikipedia.org/wiki/Echtzeit> [25.07.2011].
- Wikipedia (2011b) *Bildsynthese*. Online: <http://de.wikipedia.org/wiki/Bildsynthese> [25.07.2011].