

Die approbierte Originalversion dieser Diplom-/Masterarbeit ist an der Hauptbibliothek der Technischen Universität Wien aufgestellt (<http://www.ub.tuwien.ac.at>).

The approved original version of this diploma or master thesis is available at the main library of the Vienna University of Technology (<http://www.ub.tuwien.ac.at/englweb/>).

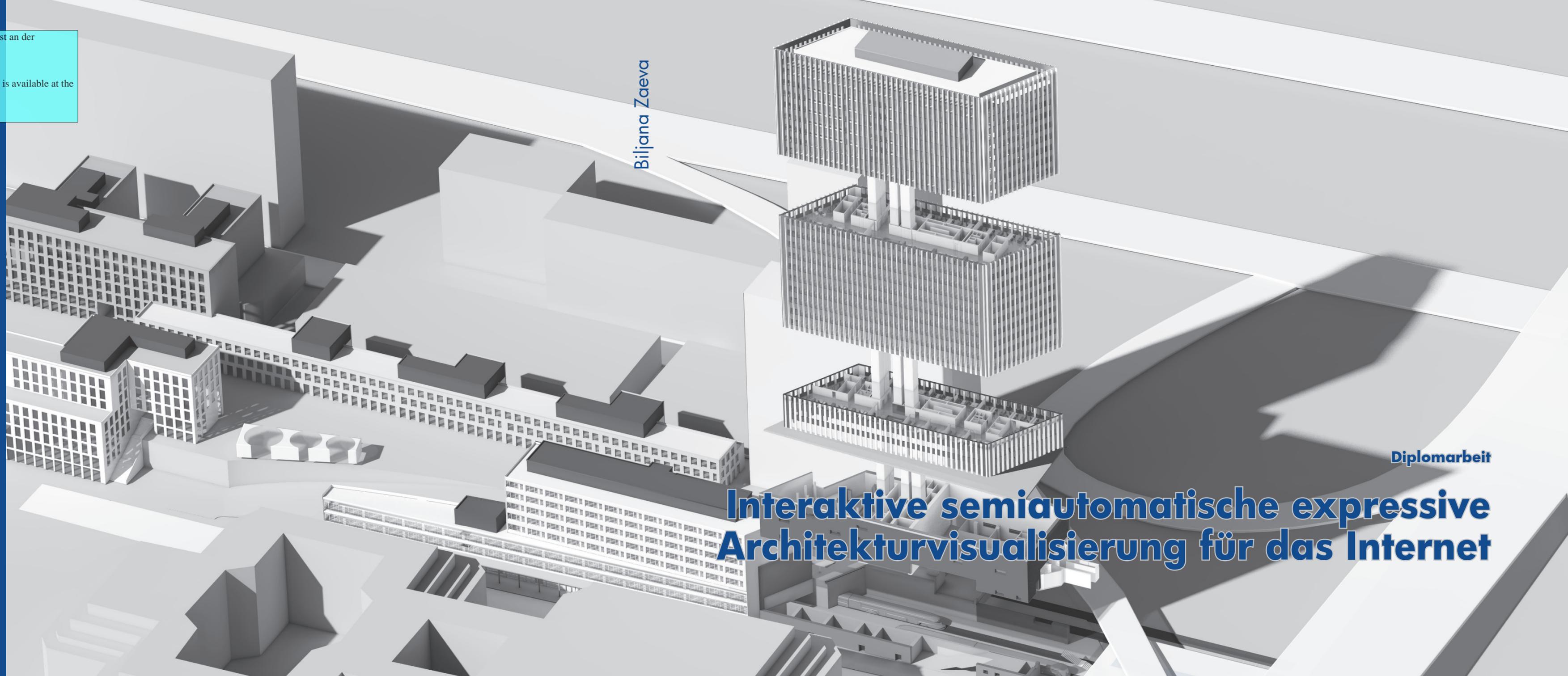
Biljana Zaeva

Diplomarbeit
Interaktive semiautomatische expressive
Architekturvisualisierung für das Internet

Zaeva Biljana
E259.1 Institut für Architekturwissenschaften
Abteilung
für Digital Architektur und Raumplanung
Technische Universität Wien

Diplomarbeit

Interaktive semiautomatische expressive Architekturvisualisierung für das Internet



Diplomarbeit

Interaktive semiautomatische expressive Architekturvisualisierung für das Internet

ausgeführt zum Zweck der Erlangung des akademischen Grades eines Diplom-Ingenieurs

Unter der Leitung von:

O.Univ.Prof. Dipl.-Ing. Dr.phil. Georg Franck - Oberaspach

Univ.Lekt. Dipl.Ing. Gabriel Wurzer

E259.1 Institut für Architekturwissenschaften
Abteilung für Digital Architektur und Raumplanung

Eingereicht an der Technischen Universität Wien

von

Zaeva Biljana

Matr. Nr.: 0227040

Salmgasse 7/16

1030 Wien

Wien, am __ November 2009

Kurzfassung/Abstract

Dreidimensionale Gebäudedarstellungen wie Architekturvisualisierung übermitteln einen guten Eindruck und Erlebnis von Räumen. Der Innenraum wird jedoch meist getrennt vom Außenraum behandelt. Weiters haben Architekturvisualisierungen oft ein Problem mit Überlagerungen. Dabei bleiben wichtige Gebäudeteile wie das Konstruktions- bzw. das bauphysikalische System versteckt. Zusatzinformationen (Bemaßungen und Beschriftungen) fehlen. Das Ziel dieser Diplomarbeit ist es, Lösungsmöglichkeiten zu untersuchen, diese sauber zu kategorisieren und eine eigene Lösungsmethode mittels eines ausgearbeiteten Beispiels zu entwickeln.

Durch Verwendung von Reorganisation, Deformation wie auch Abstraktion und Modifikation von Teilen sind Gebäuden gezielter als in realitätsnahen Architekturvisualisierungen (Renderings) präsentierbar. Die Klasse dieser Grafiken werden mangels einer anderen Bezeichnung „Expressive und Intelligente Visualisierungen“ genannt. Sie bedienen sich dreier Techniken, die schon im künstlerischen Bereich etabliert sind: Exploded Views, Cut-Aways und Ghosted Views. Neben der theoretischen Auseinandersetzung mit dem Thema wurde auch ein Beispiel (Visualisierung von TownTown) für das Web ausgearbeitet. Dieses zeigt den interaktiven Einsatz von expressiven Darstellungen in einer Architekturpräsentation. Die Interaktivität bietet den Vorteil, räumliche Zusammenhänge noch besser begreifen zu können. Innere Gebäudesysteme, wie das bauphysikalische System, die Statik, das Raumprogramm oder die Erschließung werden durch ein- und ausschaltbare Layer besser vermittelt als durch getrennt vorliegende Grafiken.

Als Zusatznutzen zur herkömmlichen Architekturvisualisierung wurde der Gewinn an Informationen durch die bewusste räumliche Änderung der Gebäudeteile, Abstraktionen und Interaktivitäten ausgewiesen. Expressive Visualisierungen können weiter für Darstellung von Wegführungen, komplexe Gebäudekonstruktionen, Immobilienvorstellungen oder Informationsvisualisierungen eingesetzt werden.

Inhaltsverzeichnis:

1. Einführung	11
2. Expressive und intelligente Architekturvisualisierung	15
2.1. Techniken expressiver und intelligenter Visualisierungen	17
• Exploded Views	17
• Cut-Aways	20
• Ghosted Views	22
• Labels	24
• Guides	25
2.2. Unterteilung von expressiven Architekturvisualisierungen in Bezug auf Interaktion	26
2.3. Unterteilung in perspektivische und axonometrische Visualisierungen	26
2.4. Stilformen für die Darstellung von interaktiven und expressiven Visualisierungen	27
• Nicht-Fotorealistische Darstellungen	27
• Fotorealistische Darstellung	29
• Einsatz von Darstellungsstilen zur Erzeugung von „Fokus und Kontext“	29

3. Verwandte Arbeiten	33
3.1. Paper: Non-Invasive Interactive Visualization of Dynamic Architectural Environments	36
3.2. Paper: Interactive Image-Based Exploded View Diagrams	37
3.3. Musikvideo „Remind Me“ und Infografik für Areva von H5	38
3.4. Webauftritt von Mckinstry	39
3.5. T-Com Haus	40
4. Entwicklung einer semi-automatischen, expressiven Architekturvisualisierung	43
4.1. Anwendungsbeispiel: Visualisierung von Town Town	43
• Allgemeine Eckdaten zum Zielgebiet	43
• Beschreibung der Visualisierung	44
• Workflow für die Implementierung	46
• Technische Implementierung	49
• Näheren Beschreibung des Plugins	51
• Nähere Beschreibung der Visualisierung	53
• Lösung der Probleme Dreidimensionaler Abbildungen in der vorgestellten Arbeit	57
5. Schlussbetrachtung	59
• Mögliche Einsatzgebiete	59
• Möglichkeiten für Weiterentwicklung	60
Anhang A: Architekturvisualisierung für das Web	63
• 2D: Vektor- und Rastergrafiken	63
• 3D: Geometrisches Modell	65
• Animation und Video	65
A.1. Technologien zur Erzeugung von interaktiven Visualisierungen im Web	66
A.2. Kriterien für die Bedienbarkeit einer Webapplikation	68

Anhang B: MaxScript und HTML/JavaScript Code	69
MaxScript Code	69
HTML / JavaScript Code	70
6. Bilderverzeichnis	75
7. Literaturverzeichnis	79

Danksagung

Besonderen Dank richte ich an Gabriel Wurzer für die hervorragende Betreuung. Seine Unterstützung, Motivation und konstruktives Feedback haben sehr zu dieser Diplomarbeit beigetragen. Weiter würde ich gern Harald Kollhammer, Karin Dantler von Soravia und Wiener Stadtwerke BMG für ihr Interesse an diesem Thema und die mir zu Verfügung gestellte Unterlagen von ihrem Bauprojekt TownTown in Wien herzlich danken. Großen Dank auch an meine Eltern, Familie und Freunde.

1. Einführung

(Problem) Um architektonische Ideen effektiv und konkret zu kommunizieren, ist es notwendig, Skizzen, Modelle oder Computervisualisierungen vorzubereiten. Bei bereits realisierten Bauobjekten sind Visualisierungen oder Modelle nützlich um sich eine Übersicht über das gesamte Gebäude zu verschaffen und komplexe Raumsysteme und Wegeführungen zu verstehen. Dreidimensionalen

Gebäudedarstellungen (Renderings) haben einige Probleme:

- Der Innenraum wird vom Außenraum immer getrennt behandelt. Es ist schwer, diese beiden mental zueinander in Beziehung zu setzen (Abb. 1).
- Es können irrelevante Details von wichtigen Teilen ablenken und diese



Abb. 1: Getrennte Darstellungen von Innen- und Aussenraum (Orestad in Kopenhagen von Daniel Libeskind)

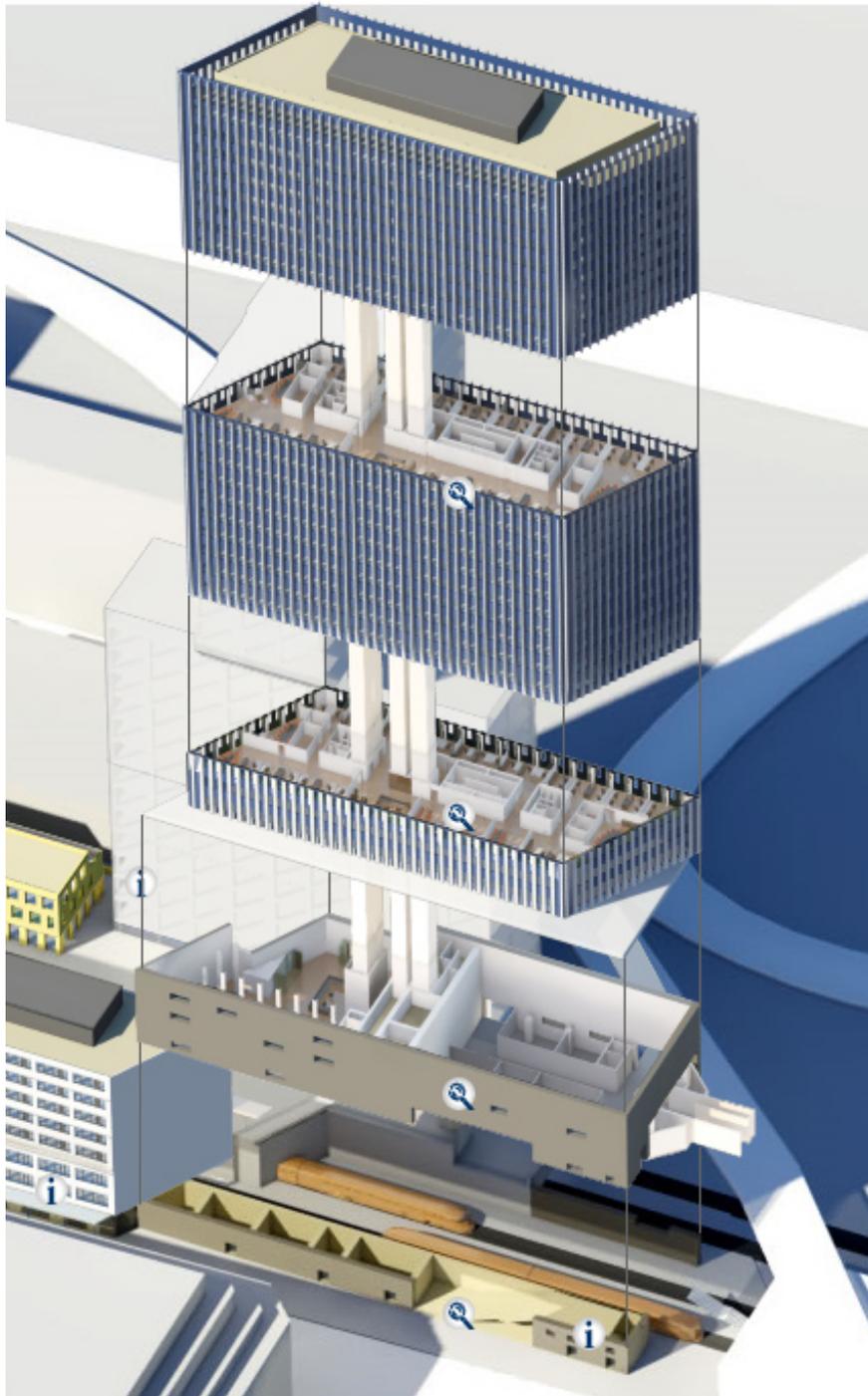


Abb. 2: Interaktive expressive Visualisierung

verstecken.

- Renderings sind gewöhnlich informationsarm. Zusatzinformationen wie Bemessungen oder Namen müssen durch ein anderes Medium ergänzt werden. Auch sind abstrakte Informationen nicht direkt ablesbar (zum Beispiel Konstruktion oder bauphysikalisches System).
- Interesse und Vorwissen der Beobachter wird nicht berücksichtigt. Es wäre wünschenswert, wenn der Benutzer selber die Informationen, die er sehen will, auswählen könnte.

Um die eben beschriebenen Probleme zu umgehen, benutzen Grafikkünstler expressive Ansätze wie Reorganisationen, Deformationen als auch Abstraktion oder Modifikationen von Teilen. Die Grafiken, bei denen im Vorhinein entschieden worden ist, was gezeigt und was ausgelassen werden soll, wurden expressive und intelligente Visualisierungen genannt¹. Die Gebäude sind in solchen expressive Visualisierungen gezielter als in realitätsnahen Darstellungen präsentiert (Abb. 2).

(Beitrag der Diplomarbeit / Aufbau) Das Ziel meiner Arbeit ist es, Möglichkeiten zu liefern, um Innenräume zu zeigen, ohne dass die gesamte Außenerscheinung des Gebäudes verloren geht. Weiters ist es mir wichtig, bedeutende Teile von Architekturmodellen in einer Visualisierung hervorzuheben und Zusatzinformationen zu integrieren. Ein Prototyp einer interaktiven Visualisierung wurde entwickelt, mit dem die Innen- und Außenräume eines Gebäudes zusammengefasst werden können. Dadurch kann ein Gebäude oder eine Liegenschaft vollständig dargestellt werden.

- Der Einsatz von expressiven und intelligenten Visualisierungen (Exploded Views, Cut Aways, Ghosted Views usw.) im Bereich der Architektur wurde in Kapitel 2 (Expressive und Intelligente Architekturvisualisierung, Seite 15) untersucht. Dabei gehe ich die Prinzipien für Reorganisation, Deformation, Abstraktion oder Modifikation von Architekturmodelle durch und gebe Wege an, mit denen Metainformationen in eine solche Visualisierung fließen können.

¹ vgl. Viola; Gröller, 2005

- Im Kapitel „Verwandte Arbeiten“ (Seite 33) zeige ich Beispiele, die mich für diese Diplomarbeit inspiriert haben oder vergleichbare Techniken und Ansätze verwendet haben.
- Die Eigenschaften einer expressiven Visualisierung werden im Zuge meiner Diplomarbeit durch ein konkretes Beispiel ausgearbeitet und getestet (Abb.2). Die ausgesuchte Plattform dieses Beispiels ist das Web. Die Möglichkeiten und die technischen Charakteristiken des Webs für eine Architekturpräsentation zeige ich im Anhang A (Architekturvisualisierung für das Web auf Seite 63).
- Das Kapitel „Entwicklung einer semi-automatischen, expressiven Architekturvisualisierung“ (Seite 43) behandelt mein Konzept und die Entwicklungsschritte der eben genannten Architekturvisualisierung durch ein Beispielmodell. Dabei teste ich diverse Features als auch Techniken für möglichst automatisches Erzeugen der Visualisierung aus.

2. Expressive und intelligente Architekturvisualisierung

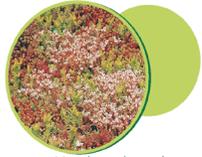
Ziel einer Architekturvisualisierung ist in erste Linie, komplexe Bauobjekte übersichtlich und verständlich darzustellen. Menschen ohne fachspezifische Kenntnisse gehören dabei oft zum Zielpublikum. Um solchen Leuten das Verständnis über eine Bauform besser zu vermitteln, werden dreidimensionale Zeichnungen (meist in Form von Renderings) erstellt, da diese im Vergleich zu 2D – Grundrissen auch ohne vorhergehende architektonische Vorbildung lesbar sind. Dreidimensionale Abbildungen haben andererseits eine Reihe von **Problemen**:

- P1: Wichtige Teile der Architektur bleiben verdeckt. Irrelevante Details verstecken und lenken von wichtigeren Teilen ab. Fotorealistische Renderings, in denen die Gebäude bis zu dem kleinsten Detail konstruiert sind, übermitteln gut einen Eindruck und Erlebnis von der Bauform.
- P2: Innenraum und Außenraum sind nicht klar zuordenbar. Es ist nicht möglich, sich ein vollkommenes mentales Bild des Bauobjekts zu bilden.
- P3: Das Konstruktionsystem ist nicht explizit sichtbar. Die Decken und Wände eines Gebäudes sind aus mehreren Schichten aufgebaut, meist sind diese wichtig um das statische System zu verstehen.

P4: Metainformationen, wie beispielsweise das Raumkonzept, werden nicht gezeigt. Weiters fehlen Zusatzinformationen wie Bemessungen oder Namen.

(mögliche Lösungen) Grafikkünstler benutzen Ansätze wie Reorganisationen, Deformationen und Abstraktion, um die beschriebenen Problematiken zu lösen. Das Ziel bleibt, bei solchen Veränderungen die relevanteste Information hervorzuheben und Objekte vollständiger darzustellen. Dabei können Details weggelassen werden, die im Kontext der gewünschten Aussage vernachlässigbar sind. Diese Visualisierungen schöpfen die menschliche Imagination aus und benutzen die Vorkenntnisse der weitgehend ungeschulten Beobachter.

Living roofs
 • Brown roofs and green planted roofs enhance the local ecology



Meadow planted



Brown roof



BREEAM
 • Targeting a rating of "Excellent" (BREEAM - Set of measurable energy efficiency and sustainability criteria)



Low water use measures
 • Low flow fixtures and fittings
 • Self closing taps
 • Leak detection facilities



Excellent transport links (walk times)
 ?? mins - Manchester Victoria Station (T, TR, B)
 10 mins - Shude Hill Metrolink and Bus Depot (TR, B)
 ?? mins - Piccadilly Gardens Bus Station (TR, B)
 ?? mins - Manchester Piccadilly Station (T, TR, B)

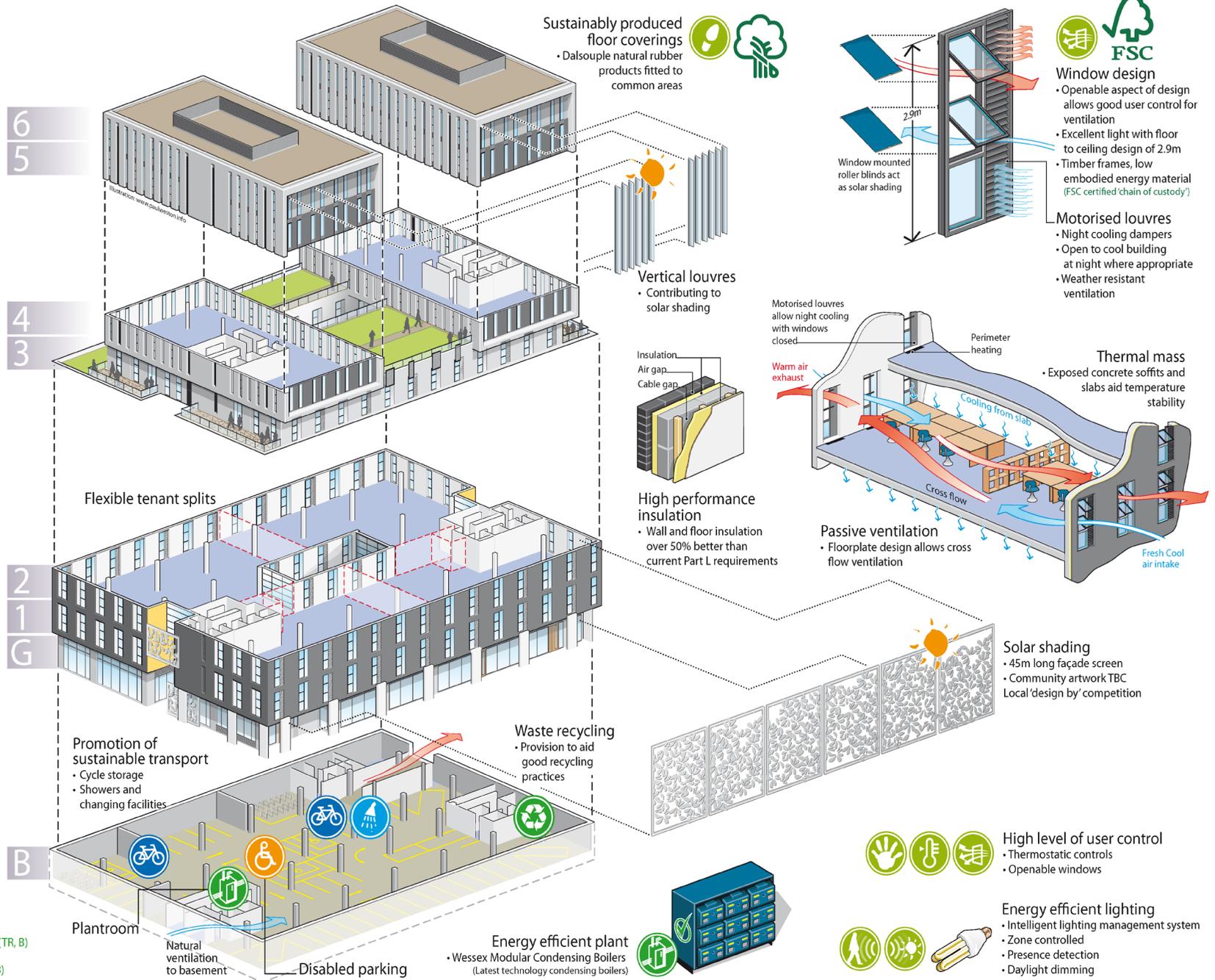


Abb. 3: Einsatz von Exploded und Ghosted View (Sustainable Workplace von Paul Weston)

(Beispiel) Als Beispiel bringen wir die Exploded View in Abbildung 3: Die Darstellungsweise ist hier nicht fotorealistisch, sondern stark vereinfacht. Die Geschosse können voneinander weggeschoben werden, um den Innenraum sehen zu können. Strichlierte Verbindungen (Guides) zwischen den Geschossen ermöglichen die mentale Zusammensetzung der Gebäudeteile. Weiters werden im Innenraum abstrahierte Personen angezeigt, um den Größenvergleich auch für ungeschulte Personen zu ermöglichen. Geschossummern werden seitlich an der Fassade angebracht. Ebenso werden Funktionsbereiche als Bezeichnung (Label) außerhalb der Fassade angebracht und gegebenenfalls mittels Strichen (Ankerlinie) verortet. Konkrete Funktionen werden mittels Symbolen angedeutet, welche direkt im Geschossinneren angebracht werden. Zusätzlich verdeutlichen semitransparent Raumteiler (Ghosed Views) die Grenzen von Nutzungsbereichen. Physikalische Prozesse wie die Luftströmung werden als Pfeile dargestellt. Herauszuhebende Details werden neben die Explosionsgrafik gestellt und wieder mittels Strichen verortet. In der Darstellung der Luftzirkulation auf der rechten Seite der Grafik wird der zu zeigende Gebäudeteil ausgeschnitten (Cut-Away View), um Platz zu sparen.

Ziel dieser Diplomarbeit ist es, die eben beschriebenen Schritte sauber zu kategorisieren und eine eigene Methode mittels der beschriebenen Verfahren zu entwickeln. Die in Verwendung beschriebenen Grafiken wurden mangels einer anderen Bezeichnung so wie im Paper „Smart Visibility in Visualization“¹ als **expressive und intelligente Visualisierungen** definiert.

2.1. Techniken expressiver und intelligenter Visualisierungen

Zu den expressiven Computergrafiken gehören Exploded Views, Cut-Aways und Ghosed Views. Diese sind schon sehr lange in der Medizin (Computertomografie, präoperative Planung) und bei technischen Illustrationen in Verwendung und zeigen effektiv die wichtigsten versteckten Informationen. In allen diesen Darstellungsweisen dienen Guides und Labels als Möglichkeit, Annotationen und Metainformationen hinzuzufügen.

¹ vgl. Viola; Gröller, 2005

Exploded Views

(Definition und Eigenschaften) Exploded Views (Abb.4) beinhalten Veränderungen der räumlichen Ordnung der Gebäudeteile. Mit anderen Worten, um wichtige Objekte visuell zu befreien, werden andere weggeschoben. Dazu wird ein Gebäude in mehrere Unterobjekte (z.B. Geschosse) aufgeteilt und anschließend werden diese voneinander verschoben, bis der Innenraum gut sichtbar wird.

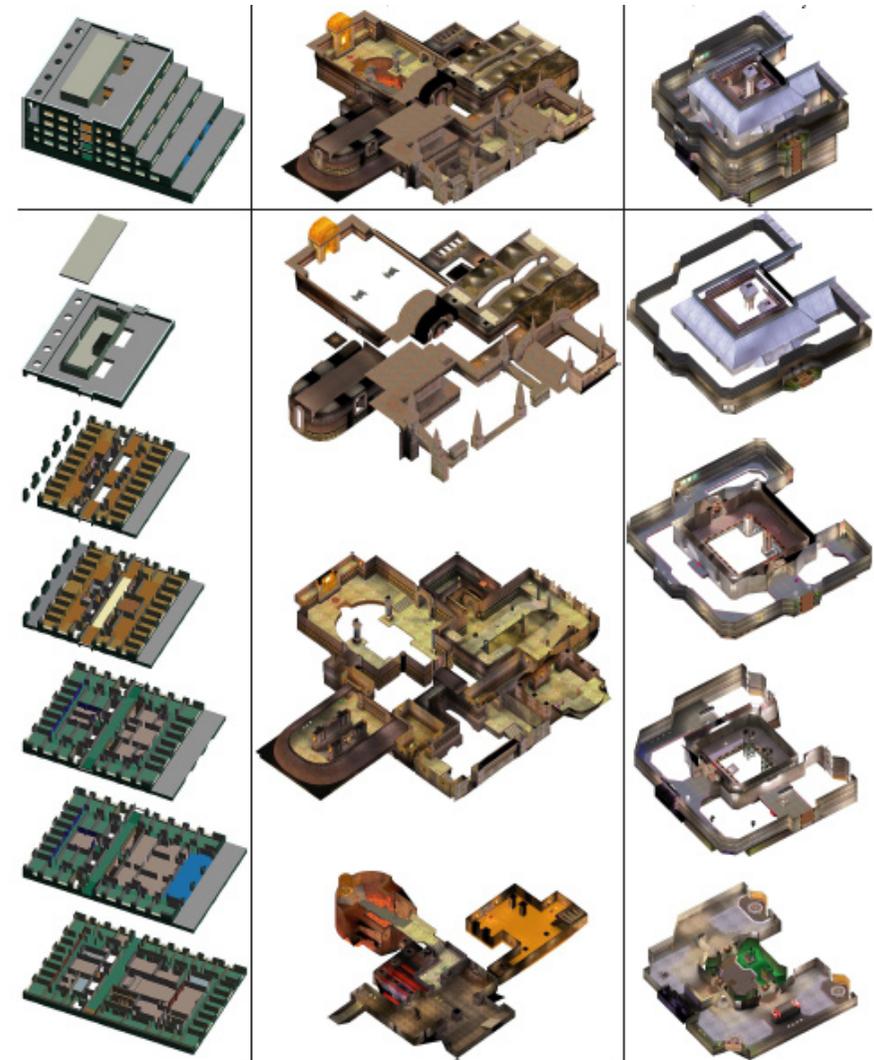


Abb. 4: Exploded View nach Niederauer et al., 2003. Drei explodierte Levels aus dem Spiel Quake III

Bei dieser Visualisierungsart werden häufig Helferobjekte wie Linien und Punkte benutzt, um die ursprüngliche Verbindung zwischen die Unterobjekte besser zu zeigen (Abb. 5). Die Vorteile dieser Visualisierungstechnik sind folgende:

- + Keine Modellteile werden beseitigt, es gehen keine Kontextinformationen verloren.
- + Ist schon seit langem die effektivste Technik, um Aufbau von Konstruktionen zu zeigen.
- + Der Aufbau der Querschnittprofile wird auch gezeigt.

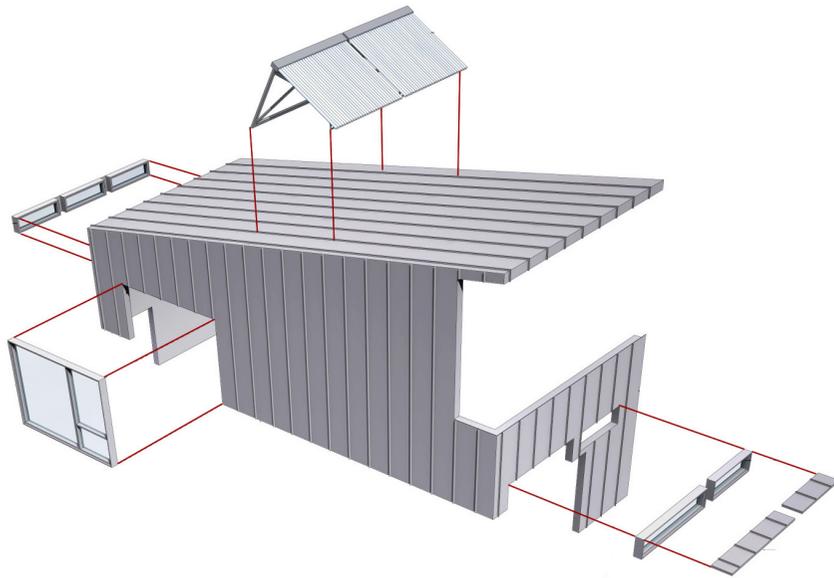


Abb. 5: Helferobjekte in eine Exploded View Architekturvisualisierung

Als erster ganz wichtiger Faktor für eine gelungene Exploded View ist die **Richtung**, in der die Teile verschoben werden, auszuwählen. Bei Architekturvisualisierungen, wo der Innenraum gezeigt werden soll, ist die vertikale Richtung üblich (Abb. 6). Aber wenn es um den Konstruktionsaufbau eines Gebäudes geht, dann werden meistens mehrere Explosionsrichtungen (Abb. 7) und Drehung von Elementen verwendet. Begrenzung der Anzahl an Explosionsrichtungen trägt zu Einfachheit der Visualisierung, leichterem Verständnis und einem richtigen Zusammenhängen der Teile bei.

(Ein-Achsen Exploded View) Bei einer einachsigen Exploded View werden alle Ebenen in der vertikalen Richtung nach oben verschoben (Abb. 6). Das Gebäude wird in Stockwerke aufgeteilt welche vertikal verschoben werden können. Bei Gebäuden mit vielen Geschossen (z.B. Hochhäuser) ergibt sich dabei ein Problem mit der perspektivischen Ansicht: Zum einen verändern die Stockwerke während der Verschiebung ihre Größe und projizierte Form, zum anderen bekommt der Betrachter eine Untersicht, sobald er das Geschoss über seinen Horizont schiebt. Deshalb muss die Kamera etwas höher als die Dachebene platziert werden. Bei Axonometrie besteht das Problem nicht.



Abb. 6: Exploded View in einer Richtung

(Drei-Achsen Exploded View) Bei komplexeren Gebäudekonstruktionen ist es notwendig die Bauelemente in unterschiedliche Richtungen zu verschieben. Die Teile sollen möglichst in Richtung der Flächennormale verschoben werden, um die Verständlichkeit beizubehalten. Ein Beispiel dafür sind Konstruktionsvisualisierungen, sie sind bei einer mehrachsigen Verschiebung besser zu zeigen bei einer einachsigen Darstellung. Als Beispiel dient die Abbildung 7: Hier werden mehreren Verschiebungsrichtungen gezeigt, wobei es sehr hilfreich wäre, den Verschiebungsweg auch grafisch anzudeuten.

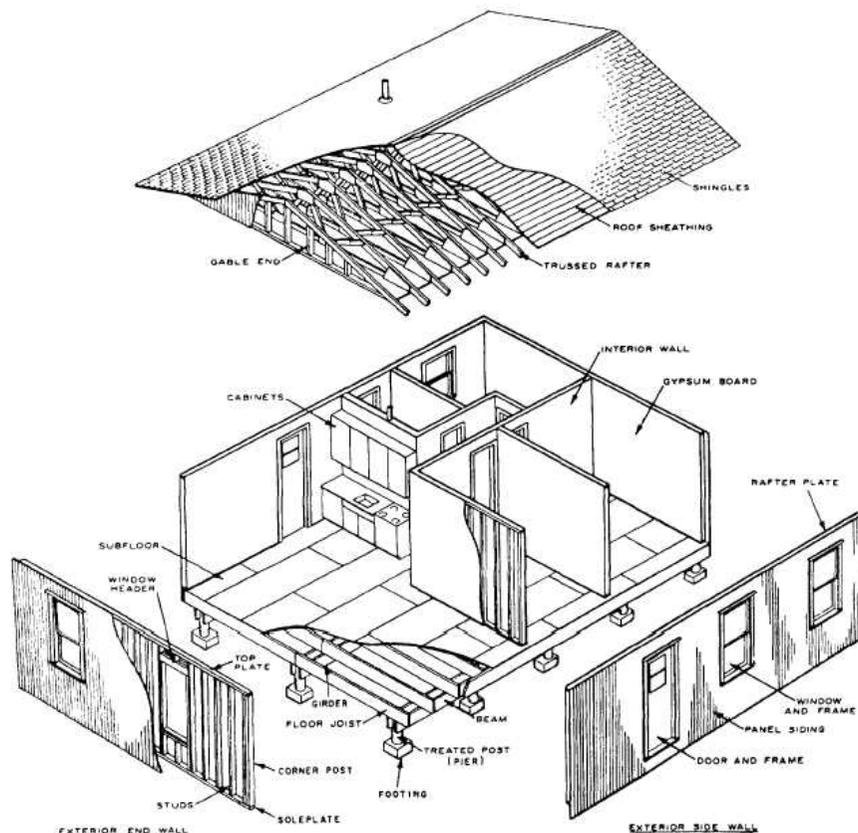


Abb. 7: Exploded View in drei Richtungen

(Gelenkige Exploded View) Bei durchgehenden Gebäudeteilen kann es sinnvoll sein, die einzelnen Teile zusätzlich zu drehen, um eine bessere Ansicht zu gewährleisten. Die Vorhangfassaden einiger Gebäude könnten zum Beispiel wie ein einzelnes äußeres Objekt betrachtet werden, dass zuerst geschnitten werden muss. Um Einsicht in das Innere zu gewährleisten, wird dieses Objekt mit einer Ebene in die Hälfte geschnitten, die Teile werden dann gelenkig voneinander weggeschoben. Um eine Drehung grafisch deutlicher zu machen, könnten kleine Symbolpfeile benutzt werden (Abb. 8).

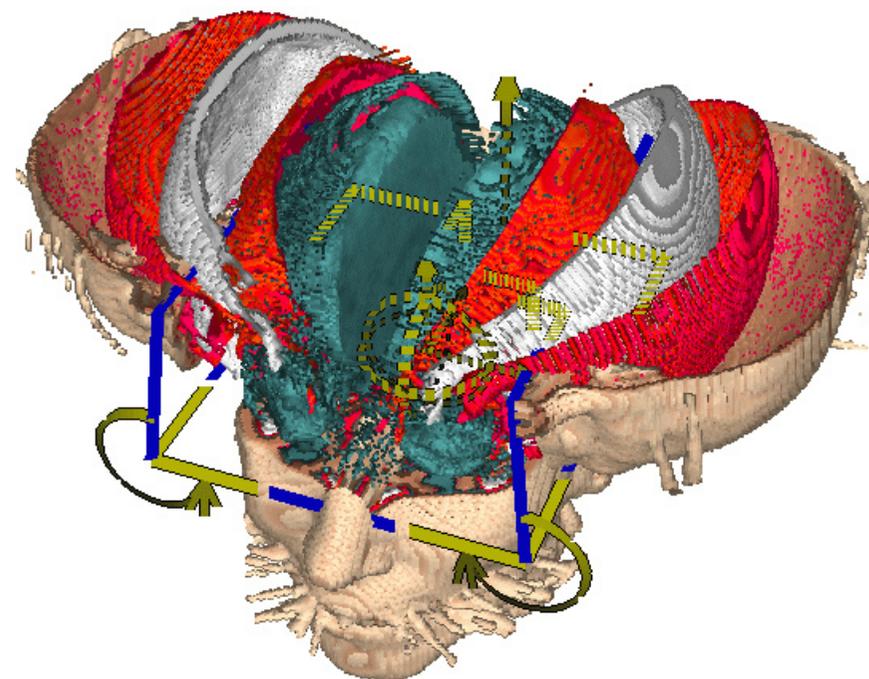


Abb. 8: Gelenkige Exploded View

(Teilexplosion) Bei der Darstellung von komplexen Bauten kann es auch oft zu einer visuellen Überfrachtung kommen: Werden diese Bauten skaliert, um genug Raum für alle Gebäudeteile auf der Bildschirmfläche zu schaffen, gehen Details verloren. Um dieses Problem zu vermeiden, können nur diejenigen Innenelemente, die wichtig sind, gezeigt werden (Abb. 9). Innenelemente, die nicht relevant sind oder sich wiederholen (zum Beispiel Regelgeschosse), könnten weiter verdeckt bleiben. Es ist auch denkbar, Explosionshierarchien zu erzeugen, in denen Teile gruppiert werden. Damit wird ein Level of Detail geschaffen: Zuerst wird der Zusammenhang des größeren Modellteils gezeigt, kleinere Elemente werden erst danach auf Anfrage des Benutzers präsentiert. Dieser Ansatz könnte bei komplexeren Gebäudekonstruktionen nützlich sein.

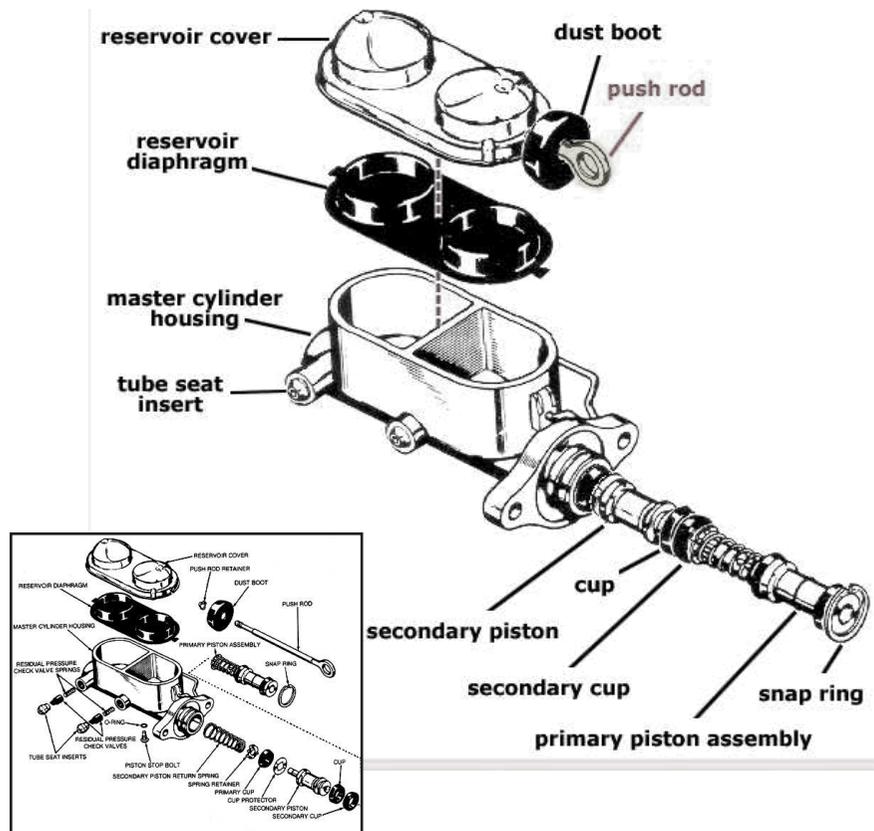


Abb. 9: Teilexplosion

Bei allen explodierten Darstellungen sollten die Einzelteile soweit von einander abgesetzt werden, dass alle interessanten Elemente zu sehen sind. Der Abstand zwischen ihnen ist so auszusuchen, dass das Modell **kompakt** bleibt. Die Elemente, die in Wirklichkeit verbunden sind, dürfen auch virtuell nicht zu weit getrennt werden². Anderes gesagt muss der Abstand zwischen jeweils zwei Ebenen so optimiert werden, kein Teil der unteren Ebene verdeckt bleibt, sowie keine großen Zwischenräume (= Platzverschwendung) zwischen den Ebenen entstehen.

Cut-Aways

(Definition) Bei Cut-Away Views (Abb.10) ist ein Teil der Wände oder Decken beseitigt, um eine Innensicht zu ermöglichen.

² vgl. Li; Argrawala, 2007

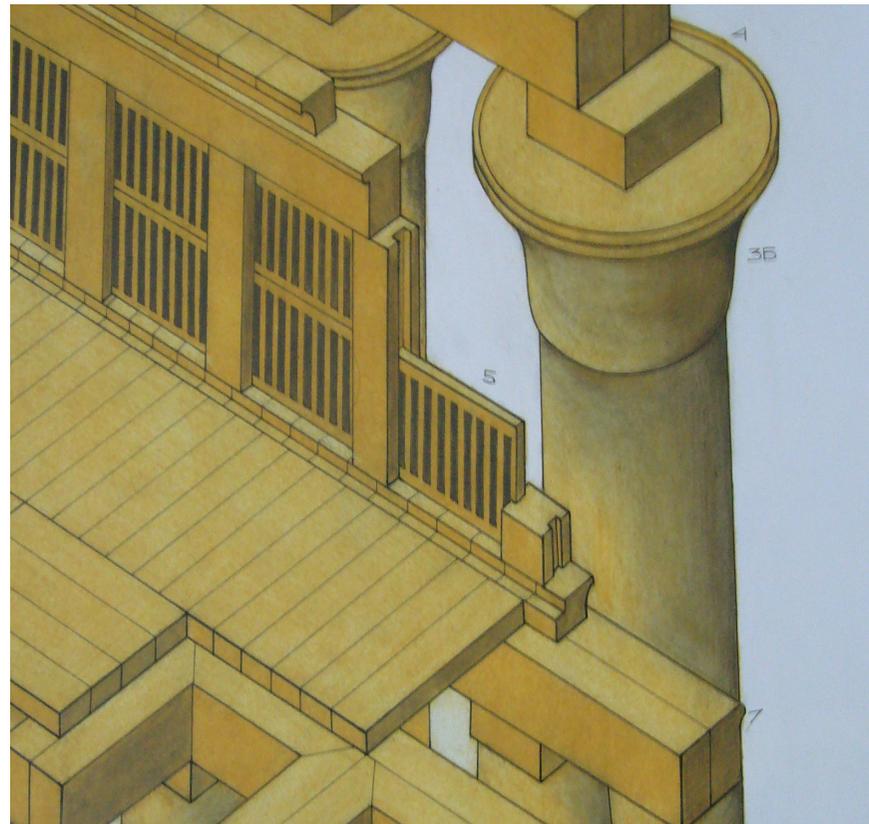


Abb. 10: Cut-Away View

Die Vorteile von den Cut-Away Views im Vergleich zu anderen bei Expressiven Techniken sind folgende:

- + Es gibt keine Doppeldeutigkeit bei der räumlichen Platzierung der einzelnen Elemente. Alle Gebäudeteile bleiben an ihrem erwarteten Platz und werden nicht im Raum verschoben.
- + Es wird ein starker Kontrast zwischen Vordergrund- und Hintergrundelementen geschaffen.
- + Der Aufbau der Querschnittprofile wird gezeigt.

Eine der wichtigsten Regeln bei Cut-Away Views ist, zu bestimmen, wo und was geschnitten werden soll. Gewöhnlich begrenzen sich Grafiker auf möglichst einfache und regelmäßige Schnittformen. In den meisten Fällen genügen eine oder zwei Schnittebenen, um alles, was notwendig ist, zu sehen. Die Position und die Orientierung von dieser Schnittebenen hängen davon ab, was gezeigt werden soll. Auch die Form der Vordergrundelemente (z.B. die Fassade) spielt eine Rolle. Trotzdem sind die Schnittebenen bei vielen Cutaway-Views an den Hauptachsen (x, y und z) orientiert und es wird entweder die Hälfte oder ein Viertel der Gebäude geschnitten. Es ist wichtig, so wenig wie möglich von dem Modell zu entfernen.

(Cutout View und Breakaway View) Es werden zwei unterschiedliche Arten von Cut-Away View unterschieden: Cutout View und Breakaway View³. Bei Cutout Views (Abb. 11) werden größere Gebäudeteile beseitigt. Bei der Breakaway Views (Abb. 12) werden nur kleinere Löcher an den Vordergrundelementen gemacht. Cutout Views sind nützlich, wenn mehrere Objekte oder größere Innenräume gezeigt werden sollen. Im Unterschied dazu werden bei Breakaway Views wenige, kleinere Innenobjekte präsentiert. Dabei sollte das ausgeschnittene Loch genau so groß wie die wichtigen innenliegenden Objekte sein. Wenn Cut-Away Views als 3D Modell benutzt werden, dann ist wichtig, dass der Ausschnitt groß genug ist, um die Innenobjekte von einem höheren Ansichtswinkel aus zu sehen.



Abb. 11: Cutout View



Abb. 12: Breakaway View

3 vgl. Diepstraten; Weiskopf, 2003

Es bestehen mehrere Möglichkeiten, die **Abgrenzung des Innen- und des Außenraums** hervorzuheben. Am besten wird das mit der Abstraktion der Schnittlinien gemacht. Die geschnittene Fläche kann mit einer Farbe ausgefüllt werden (Abb. 13), die Randlinien stärker hervorgehoben oder die geschnittenen Schichten dunkler oder in einer anderen Farbe als das restliche Modell dargestellt werden. Die Schnittlinie muss auch nicht immer gerade verlaufen. Ein Schnitt mit Treppeneffekt trägt zu größerem Differenzieren zwischen Innen und Außen des Bauobjekts bei. Wenn der Schichtenaufbau an der Schnittlinie größere Bedeutung hat, um zum Beispiel einen Deckenaufbau zu zeigen, dann sind die Elementen unterschiedlich lang zu schneiden.

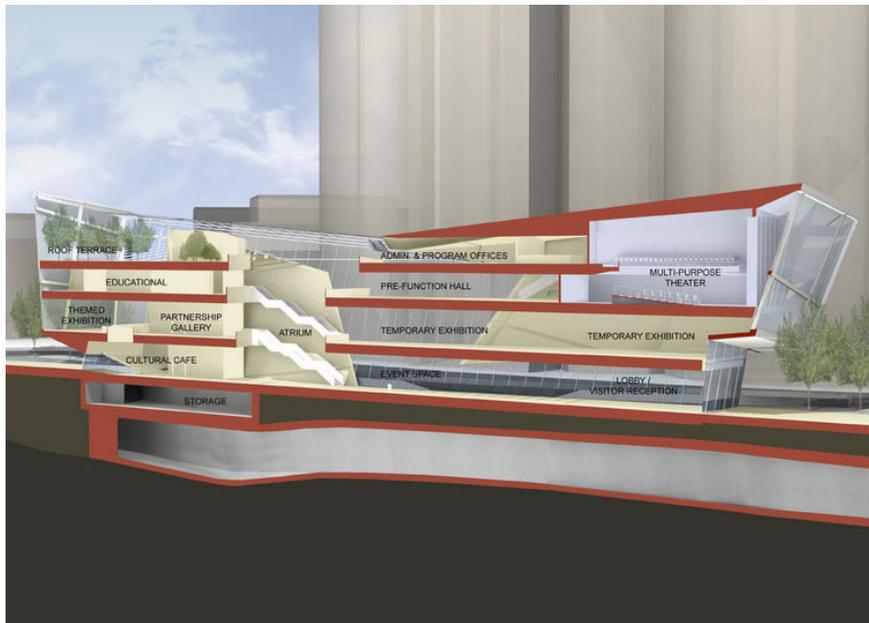


Abb. 13: Schnittlinien bei Cut-Away View
(Center for Arts and Culture in Boston von Daniel Liebeskind)

Ghosted Views

(Definition) Ghosted Views (Abb. 14) zeigen die Interessensgebiete im Innenraum durch Transparenz der vorne liegenden Gebäudeelemente.



Abb. 14: Ghosted View

Ihre größten Vorteile gegenüber anderen expressiven Visualisierungen sind folgende:

- + Es gibt keine Doppeldeutigkeit bei der räumlichen Platzierung der einzelnen Elemente. Alle Gebäudeteile bleiben an ihrem erwarteten Platz und werden nicht im Raum verschoben.
- + Sie zeigen räumliche Tiefe besser, weil mehrere überlappende Objekte dargestellt werden.

Das wichtigste Thema bei Ghosted Views ist die **Transparenz**. Das Wort kommt von der lateinische Sprache (trans - „(hin)durch“, parere „sich zeigen, scheinen“) und damit wird eine optische Eigenschaft eines Materials bezeichnet⁴. Im Allgemeinen ist ein Material transparent oder durchsichtig, wenn Dahinterliegendes relativ klar zu erkennen ist. Die Transparenz bei Ghosted Views wird verwendet, um Objekte anzuschauen, die normalerweise von undurchsichtigen verdeckt sind. Deshalb sollte diese nicht durchgehend verlaufen, sondern nur vor den hinterlegten Objekten, die gezeigt werden sollen, erhöht werden (Abb. 15). Dabei bleibt der Rest semitransparent bis zu ganz

⁴ vgl. <http://de.wikipedia.org/>, 2009

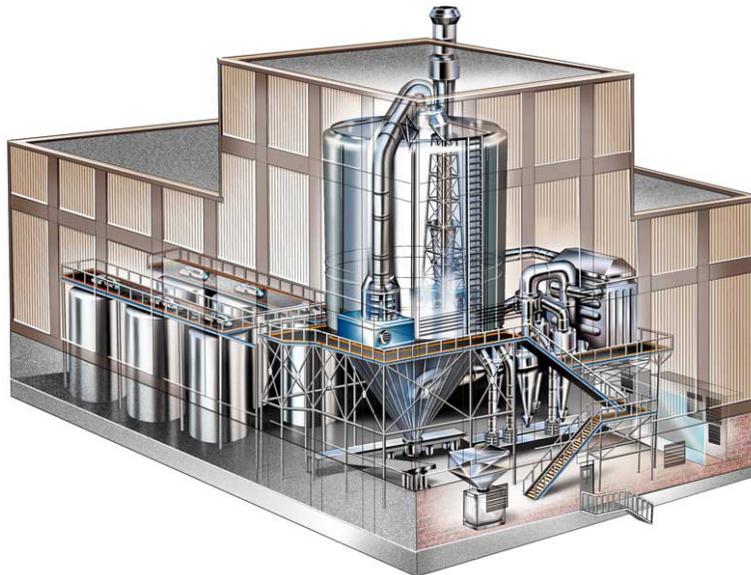


Abb.15: Ghosted View

undurchsichtig.

(Sonderform Phantom Views) Eine Sonderform von Ghosted Views sind die Phantom Views. Hier werden nur die Konturen der äußeren Elemente dargestellt (Abb. 16), was die simpelste Art der Implementierung von Ghosted Views darstellt. Diese Konturen werden in einem anderen Stil als die hinterlegten Elemente gerendert, um den Unterschied eindeutiger zu machen. Jedoch gehen bei Phantom Views die Details der konturgezeichneten Objekte verloren (Nachteil). Auch wird das Material und die Eigenschaften der Oberfläche ignoriert. Im Grunde genommen wird in diese Methode auf zwei Formen der Sichtbarkeit reduziert: realistisch dargestellte und konturgezeichnete Elemente⁵. Wegen diese Nachteile empfiehlt sich der Einsatz von Phantom Views nur in Sonderfällen.

⁵ vgl. Diepstraten; Weiskopf, 2002



Abb. 16: Phantom View

Labels

(Definition) Mit Labels sind Text- und Bildetiketten, sprich: die „Benennung“ von Teilen eines Objekts einer Architekturvisualisierung gemeint. Sie sind ein essenzieller Teil von technischen Zeichnungen und helfen bei der Erklärung und Kontextualisierung der Bilder⁶. Zusätzlich zu Labels werden im Bild oft auch noch Legenden oder Bildüber- und Unterschriften verwendet. Mit ihrem Einsatz entsteht eine Wechselbeziehung zwischen visuellen und textuellen Informationen, beziehungsweise werden technische Begriffe, Bezeichnungen und weitere Erklärungen hinzugefügt.

6 vgl. Tufte, 2006

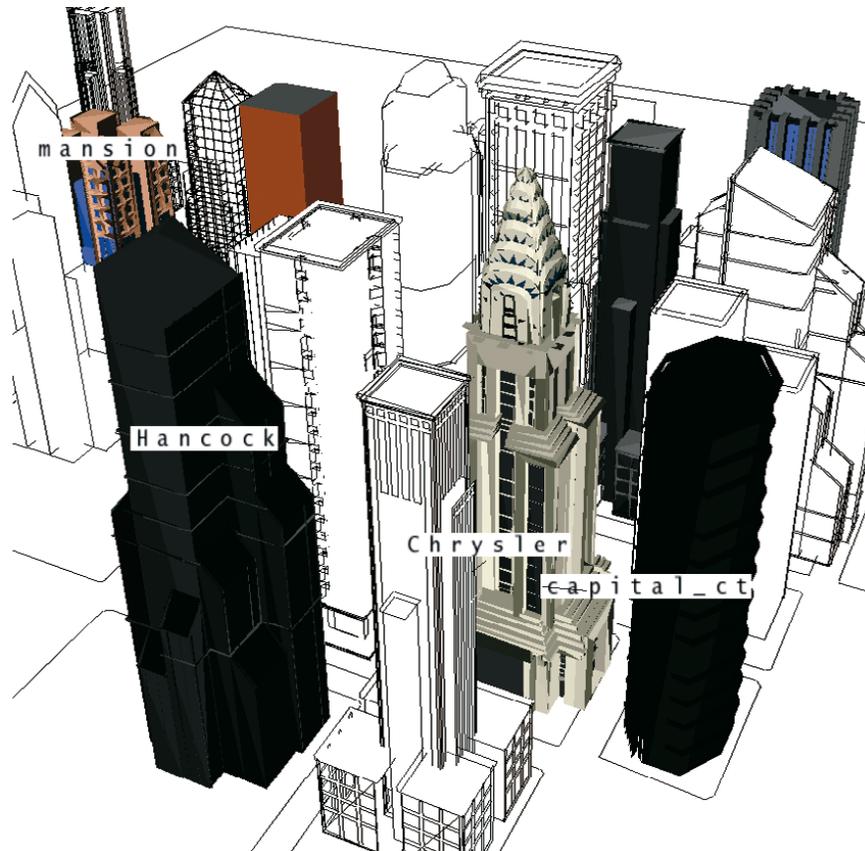


Abb. 17: Internal Labels

Die Labels sind vorsichtig zu positionieren, um eine klare visuelle Korrespondenz zwischen dem Text und der verknüpften Region des Modells zu schaffen. Abhängig davon, ob ein Label direkt auf dem Objekt platziert ist oder außerhalb, ist zwischen internen (Abb. 17) und externen (Abb. 18) Labels zu unterscheiden. Die internen Labels überlagern das Objekt direkt. Dabei sollte sichergestellt werden, nicht zu viele bedeutende Objektinformationen damit zu verdecken. Deshalb sind diese nur dort zu verwenden, wo es sich um wenige kleine Labels oder große Objekte handelt. Im Gegensatz dazu verwenden externe Labels die Aussenfläche der Grafik zur Anzeige eines Texts oder Bildes, und verankern sich

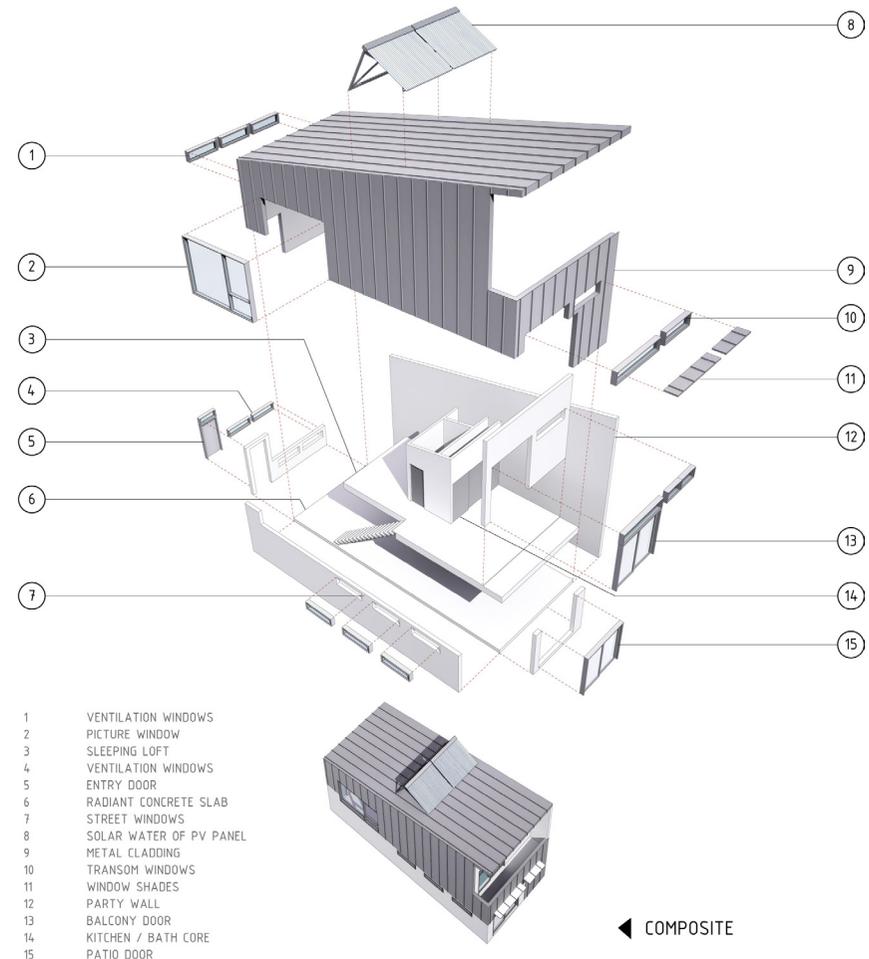


Abb. 18: Externe Labels mit zusätzlicher Legende

über eine Linie (Ankerlinie) mit einer bestimmten Position im Bild (Ankerpunkt). Der Ankerpunkt kann grafisch hervorgehoben werden, damit wird das externe Label besser identifizierbar.

(Layout von Labels) Um ein effektives Layout für Labels zu erzeugen, sollten einige Anforderungen erfüllt werden: Die Labels sollten groß genug gewählt werden um lesbar zu sein. Überdeckung zwischen Labels sollte ausgeschlossen werden, dies kann durch Einsatz von geeigneten Algorithmen (map labeling, graph layout optimisation) bewerkstelligt werden. Des Weiteren sollte klar sein, welche Bildteile durch welche Labels erklärt sind. Um das zu erreichen können die Ankerpunkte grafisch hervorgehoben und die Distanz zwischen Ankerpunkt und Label so klein als möglich gehalten werden. Für Ankerlinien gilt, dass diese stets überkreuzungsfrei gezeichnet werden sollen. Auch ist zu entscheiden, ob Ankerlinien durch gerade oder orthogonale Linien (entlang den Hauptachsen ausgerichtet oder geknickt) dargestellt werden sollen.

Guides

(Definition) Guides sind Linien die Beziehungen zwischen den verschiebbaren Elementen einer expressiven Visualisierung verdeutlichen. Abbildung 19 zeigt eine Explosionsgrafik, in der die Geschosse entlang der vertikalen Erschließung durch strichlierte Linien in Beziehung gesetzt werden. Eine solche Darstellung kann auch zusätzliche Elemente enthalten, welche zum Beispiel in Form von Pfeilen die Explosionsrichtung angibt (Abb. 20).



Abb.19: Verwendung von Linien

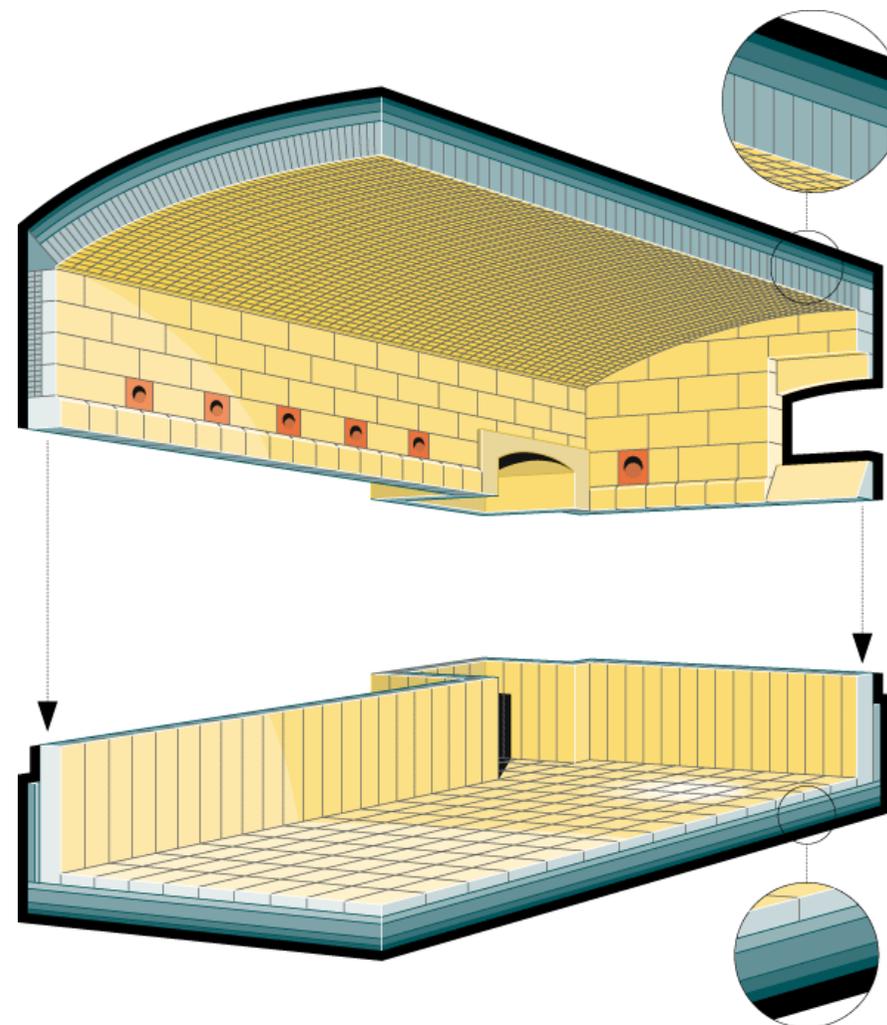
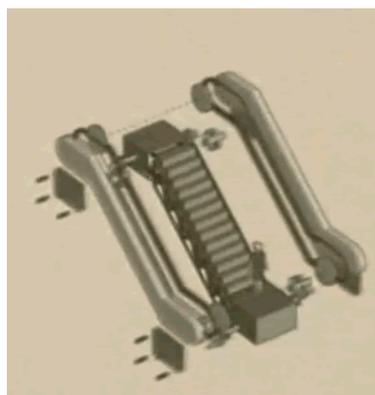


Abb. 20: Exploded View mit Pfeile und eingezoomte Detailelementen

2.2. Unterteilung von expressiven Architekturvisualisierungen in Bezug auf Interaktion

Bei einer Architekturvisualisierung kann man grundsätzlich drei Stufen der Interaktivität unterscheiden: Statische Grafiken, Grafiken mit einfacher Interaktionsmöglichkeit und Grafiken die eine Echtzeitnavigation zulassen. Die Unterteilung ist für die spätere technische Umsetzung in Form einer eigenen Visualisierung sinnvoll.

(Statische Grafiken) Die statischen expressiven Darstellungen sind zweidimensionale Zeichnungen. Die meisten werden von 3D Modellen erzeugt. Wegen dem geringen Speicheraufwand, der schnellen Erzeugbarkeit und der Möglichkeit, die Methode ohne extra Software benutzen zu können, sind solche Darstellungen sehr verbreitet. Der Nachteil statischer Grafiken ist, dass das unveränderte (nicht explodierte) Gebäude niemals zu sehen ist. Hier wäre es vorteilhaft, für ein besseres Verständnis eine zusätzliche Zeichnung vom Gebäude an der Seite zu zeigen.

(Grafiken mit einfache Interaktionsmöglichkeiten) Einfache Interaktionsmöglichkeiten reduzieren sich meistens auf das Ein-/Ausmachen von bestimmten Elementen, also Veränderung der Sichtbarkeit von Teilen der Zeichnung. Meistens kann dabei zwischen Explosion und Originalform hin- und hergeschaltet werden, daneben können Beschriftungen an- und abschaltbar sein. Der Nachteil bei dieser Methode, wie auch bei den statischen Zeichnungen, ist, dass immer eine Ansicht der Gebäude nicht zu sehen ist. Wenn es sich um ein komplexeres Bauobjekt handelt, ergibt sich das Problem der Überlagerung (Cluttering): Viele Informationen sind auf einen zu kleinen Raum darzustellen, wobei eventuell noch Bauteile verdeckt bleiben können.

(Grafiken mit Interaktiver Echtzeitnavigation) Bei diesen Darstellungen wird das gesamte 3D Modell ausgearbeitet und den Benutzern zur Manipulation gegeben. Das Modell kann dabei direkt untersucht werden durch Drehen oder Ein-/Auszoomen. Durch eine Navigation sind meistens zusätzliche Kontrollmöglichkeiten vorhanden. Ein Nachteil der

Echtzeitnavigation ist der große Aufwand der Aufbereitung des Modells sowie der Interaktionsfunktionalität. Diese Präsentationsmethode ist auch speicherbelastend, es kann zu Wartezeiten und Verzögerungen kommen falls die Hardware nicht entsprechend ausgelegt ist. Praktisch gesehen ist diese Darstellungsform in der Architekturpräsentation im Web selten zu finden.

2.3. Unterteilung in perspektivische und axonometrische Visualisierungen

Axonometrie und Perspektive sind Projektionstechniken. Bei einer **Perspektive** werden die Objekte so abgebildet, wie sie dem Auge des Betrachters von einem bestimmten Punkt aus erscheinen. Dabei ist die Verkürzung der Objekte dargestellt und es entsteht ein korrekter räumlicher Eindruck (Abb. 21b). Zur Herstellung perspektivisch richtiger Zeichnungen werden Fluchtpunkte benutzt. Mit dem Begriff Parallelperspektive werden die **Axonometrie** und die scharfe Parallelprojektion (Militär- und Kavalierperspektive) bezeichnet. Axonometrie ist ein allgemeiner Begriff für Isometrische, Diametrische und Trimetrische Projektionen (Abb. 21a). Eine der wichtigsten Eigenschaften der Axonometrie ist die nicht hierarchische Behandlung von Räumen. Die Objekte im Vordergrund haben dieselbe Größe wie die Hintergrundobjekte. Die Objektform bleibt dabei ohne Deformationen und erhält die Größe des parallel projizierten Objekts.

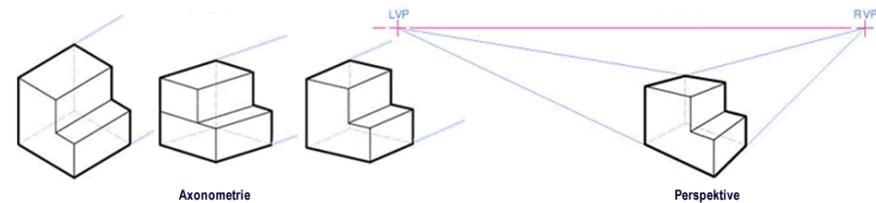


Abb. 21: a) Axonometrie; b) Perspektive

(Verwendung in der architektonischen Visualisierung) Architekten haben lange hauptsächlich Perspektive für dreidimensionale Gebäudeaufzeichnung verwendet. Erst Anfang des 19. Jahrhunderts haben einige Architekten wie Gropius oder Le Corbusier bewusst zu der isometrischen und militärischen Projektion als Werkzeug gegriffen⁷. Der Vorteil dieser perspektivischen Zeichnungen ist die korrekte Gebäudeform und -größe. Diese übermitteln eine objektive Ansicht der Gebäude ohne Verzerrungen. Andererseits behandeln die perspektivischen Zeichnungen den Eindruck beziehungsweise das sichtbare Bild von einem Gebäude. Dabei ist aber das Gebäude von einem subjektiv ausgewählten Blickwinkel dargestellt. Abhängig vom Standpunkt der Betrachter entstehen unterschiedliche Verzerrungen, die das objektive Verständnis der Form beeinträchtigen können. Für interaktive und expressive Grafiken wird meist die Axonometrie verwendet. Dies liegt an ihre Nähe zu technischen Illustrationen, und nichts anderes sind Exploded Views, Ghosted Views und Cut-Aways. Nur bei der Visualisierung von kleineren Strukturen können perspektivische Darstellungen in solchen Grafiken beobachtet werden. Diese sind meist statisch, eine dynamische interaktive Visualisierung mit Perspektive ist fast nicht zu finden.

2.4. Stilformen für die Darstellung von interaktiven und expressiven Visualisierungen

Bei einer Architekturdarstellung sind mehrere visuelle Umsetzungen möglich. Die Objekte können bewusst möglichst **realistisch** oder in unterschiedlichem Maß **abstrahiert** werden. Es sind Entscheidungen darüber zu treffen, welcher gestalterische Stil für den konkreten Fall geeignet ist. Alle Darstellungstechniken können in zwei Gruppen zusammengefasst werden: Nicht -Fotorealistische- und Fotorealistische Darstellungen. Die beschriebenen Stile können anschließend dazu verwendet werden, um eine „Fokus und Kontext“ Darstellung im expressiven Visualisierungen zu erzeugen.

⁷ vgl. Kato, 2007

Nicht-Fotorealistische Darstellungen

Beim Nicht-Fotorealistischen Darstellungen geht es hauptsächlich darum, ein Modell künstlerisch ausgeben zu lassen. Dazu zählen Techniken, die direkt von der Kunst inspiriert sind (zum Beispiel „Bleistift“ Darstellungen, Cartoon Shading) wie auch solche, die sich auf Abstraktion spezialisieren (stilisierte Darstellungen).

(„Bleistift“ Darstellungen) Bei einer solcher Technik werden die Objekte künstlerisch dargestellt. Dabei sind die Materialeigenschaften komplett abstrahiert (Abb. 22).

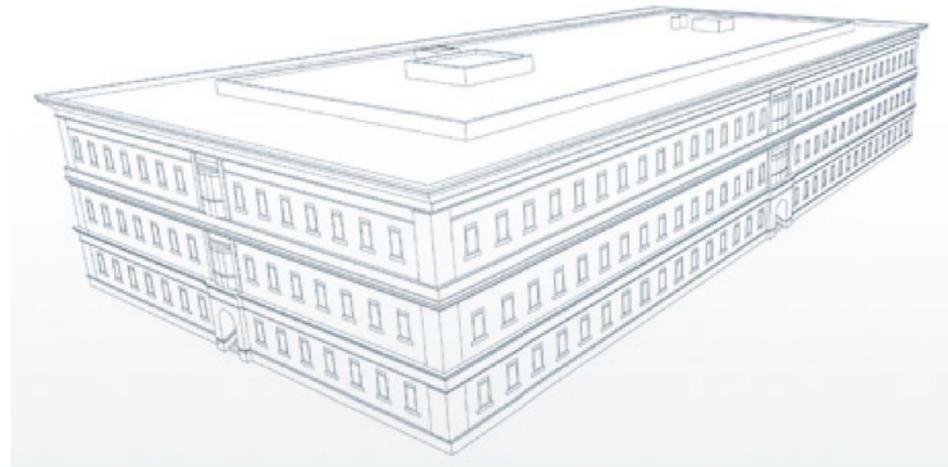


Abb. 22: Bleistift Darstellungen

Der Vorteil dieser Darstellungsmethode ist der große Kontrast, wobei die Objekte gut sichtbar bleiben. Als Nachteil können in manchen Fällen schlechtere räumliche Darstellung und das Fehlen von Tiefeninformation gesehen werden. Die räumliche Wahrnehmung könnte teilweise durch Benutzen von unterschiedlichen Linienstärken verbessert werden. Mehr Detailinformationen können durch zusätzliche Schraffurtechniken wie Hatching oder Strippling hinzugefügt werden⁸.

⁸ vgl. Viola, 2005

- **Hatching:** Gezeichnete Flächen mit der Hatching Schraffurtechnik übertragen gleichzeitig Materialeigenschaften, Farbton und Objektform. Der Begriff Hatching bezieht sich auf eine Gruppe von Strichen mit räumlich kohärenter Richtung und Eigenschaften. Hatching ist bei Entwurfszeichnungen sehr geeignet und imitiert handgezeichnete Bleistiftzeichnungen. Verschiedene Strichtexturen können benutzt und in einer 3D Textur zusammengefasst (Abb. 23) werden. Die einzelnen Texturen sollen in ihrer Form sehr ähnlich sein, um fließende Übergänge zu ermöglichen.

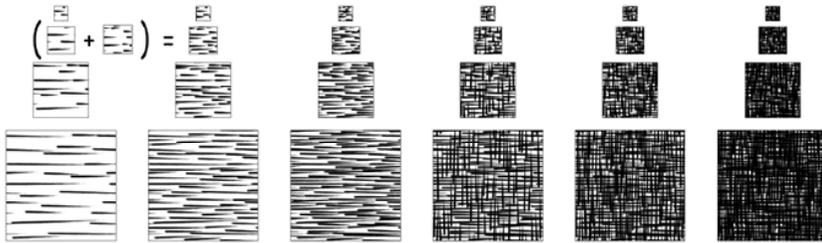


Abb. 23: Hatching Straffuren

- **Stripling:** Äquivalent zur Kunstform werden bei Stripling Objekte durch viele kleine Punkte dargestellt. Diese Technik findet meist Anwendung bei wissenschaftlichen und technischen Illustrationen. Ein Stripling mit der Hand herzustellen ist allerdings außerordentlich aufwendig, da oft mehrere tausend Einzelpunkte gesetzt werden müssen. Der richtige Ton der Fläche und die Oberflächeneigenschaften werden durch die Punktestärke und Dichte dargestellt (Abb. 24).

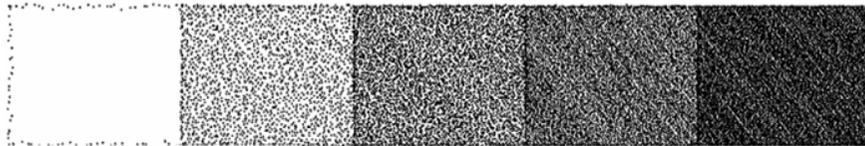


Abb. 24: Stripling Straffuren

(Cartoon Shading) Ziel dieser Darstellungsart ist, das Modell wie bei einer vereinfachten technischen Zeichnung aussehen zu lassen (Abb. 25). Die Schattierung der Szene ist meistens nur sehr einfach gehalten und es gibt wenige Abstufungen pro Farbe. Als Beleuchtungsmodell kommt dabei oft das Gooch Schattierungsmodell zum Einsatz, welches Farbintensitäten als kalt/warm- Skala darstellt. Die Umrisse (Silhouette) eines Objekts können je nach gewünschtem Effekt gezeichnet werden.

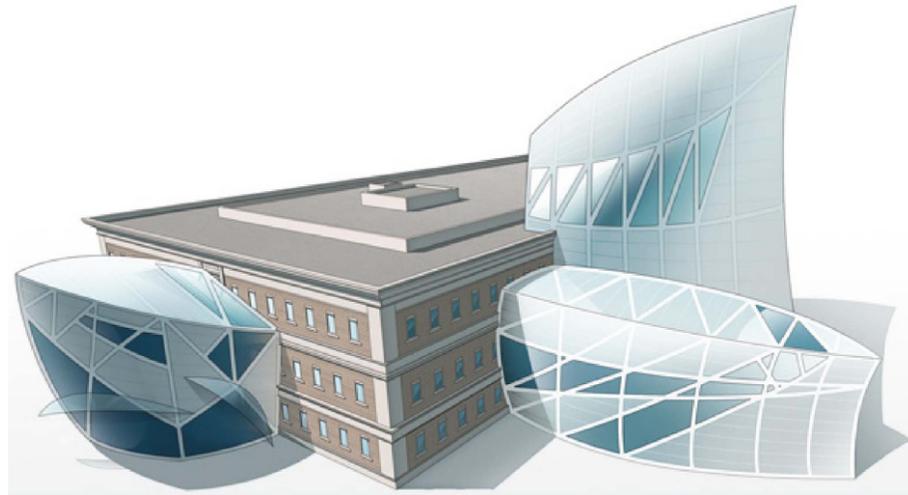


Abb. 25: Cartoon Rendering

(Stilisierte Abstraktion) Es gibt eine Vielzahl an automatischen Algorithmen, die ein 3D Modell simplifizieren können⁹. In der Praxis mag es jedoch sinnvoll sein, manuell eine zweite Version eines Modells zu erzeugen, weil damit die Kontrolle über die Geometrie in Händen des Architekten behalten werden kann.

⁹ vgl. Hoppe, 1996

Abstraktion stellt weniger eine vordefinierte Darstellungstechnik, als vielmehr ein Mittel dar, mit dem wichtige Teile eines Modells hervorgehoben oder verstärkt werden können. Sie ist deshalb in der NPR anzusiedeln, weil solche Techniken schon seit langem in der Kunst gebräuchlich sind.

Fotorealistische Darstellung

Fotorealistische Darstellung ist eine Darstellungsart die genau ist und keinen Raum für Interpretation lässt. Die Gebäude sind detailliert ausgearbeitet und die Eigenschaften der Materialien werden naturtreu repräsentiert. Um solche realistischen Bilder zu erstellen, ist es wichtig, sich erst Klarheit über die physikalischen Prozesse der Lichtausbreitung zu verschaffen. Beleuchtungswerte werden in der Architektur vor allem für die Beleuchtungsplanung oder Tageslicht- und Sonnenlichtsimulationen angewendet. Aber auch bei der wirklichkeitstreuen Darstellungen von Architektur in Form von Renderings ist Fachwissen rund um Beleuchtung mitzubringen. Realistische Visualisierungen können durch ein hohes Maß an ausgearbeiteten Details zu einer Überlastung der Betrachter führen. Einige unwichtige Informationen können von den wichtigen Details ablenken oder diese verdecken. Der Erstellungsaufwand einer fotorealistischen Darstellung ist groß im Vergleich zu den anderen eben präsentierten Darstellungsstilen.

Einsatz von Darstellungsstilen zur Erzeugung von „Fokus und Kontext“

(Definition mit Beispiel) Expressive Visualisierungen heben oft eine bestimmte Eigenschaft, einen Prozess oder ein Element als Fokus der Visualisierung hervor. Um das zu erzeugen werden andere Eigenschaften, Prozesse oder umgebende Elemente abstrahiert (Kontext Informationen). Das oft gesehene Beispiel für Fokus/Kontext Visualisierungen in der Architektur ist ein Umgebungsmodell oder –Rendering (Abb. 26). Die Nachbargebäude und die Wegführung werden in eine Visualisierung mit unterschiedlichen Renderingstilen, Farbmodellen oder Beleuchtungsmodellen dargestellt und damit grafisch zu weniger relevant eingestuft¹⁰. Diese Kontextelemente sind beizubehalten, weil sie zusätzliche Informationen liefern und räumliche oder referentielle Beziehungen bieten.

¹⁰ vgl. Viola, 2005

Um die Fokuginformation noch mehr in Vordergrund zu ziehen, sollten die Kontextelemente so dargestellt werden, damit sie weniger Bildfläche besetzen. Es gibt mehrere Techniken, die bessere Ausdifferenzierung des Fokus und des Kontexts zu ermöglichen. Manche Techniken können auch unterschiedlich kombiniert werden.

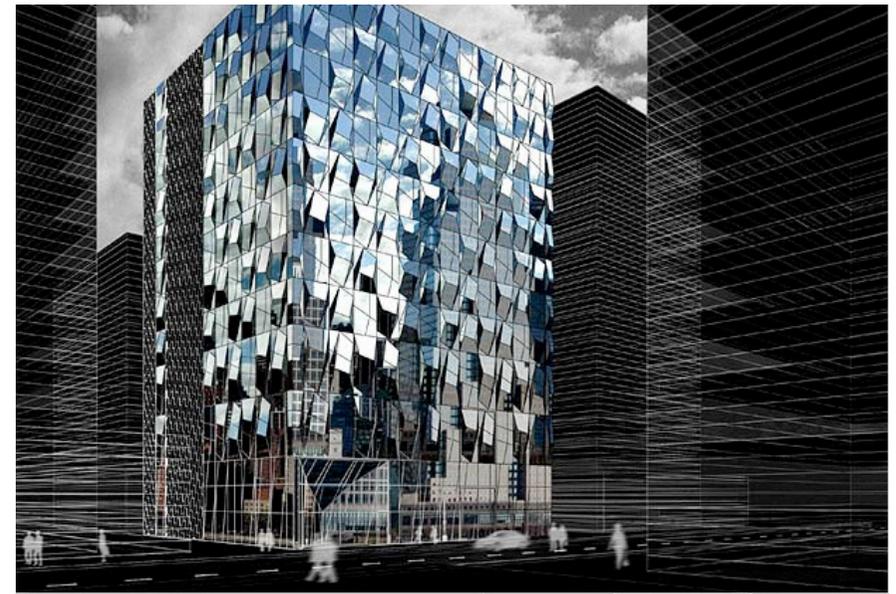


Abb. 26: Umgebungsrendering (Trutec Building in Seoul von Barkow Libinger Architects)

(Einsatz von kontrastierenden Renderingstilen) Renderingstile wie zum Beispiel die „Bleistift“ Technik sind sehr gut für Repräsentation des Kontexts einer Visualisierung geeignet. Dabei können Elemente in unterschiedlichem Ausmaß abstrahiert werden. Das Fokuselement wird hingegen durch realistische Schattierung betont (Abb.26). Kontrastierende Renderingstile sind auch mittels anderer Techniken denkbar, das Grundmuster bleibt jedoch erhalten.

(Abstraktion) Fokuselemente können detailliert ausgearbeitet werden, wobei der Kontext nur grob gezeichnet wird (Abb. 27). Die räumlichen Informationen der Kontextelemente werden abhängig vom Modellmaßstab, der Aufgabe und dem Benutzer reduziert.

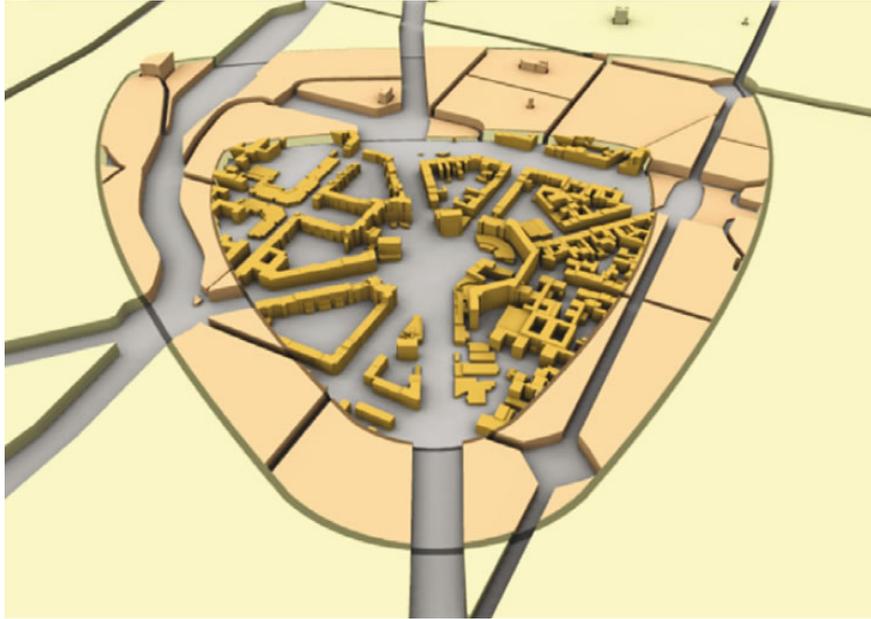


Abb. 27: Detailstufen in Stadtmodelle

(Semantische Tiefenschärfe) Diese Methode ist eine wortwörtliche Interpretation des Begriffs Fokus, wobei ein einziger Bereich scharf beibehalten werden soll und alles, was weniger relevant ist, verschwommen dargestellt wird (Abb. 28). Bei Kameras bezeichnet der Begriff „Tiefenschärfe“ auf die Distanz des Objekts zur Fokusebene. Der Blick des Betrachters wird durch Einstellung der Blende, der Entfernung und der Brennweite auf das Hauptmotiv gelenkt. Der Vorder- und Hintergrund wird dadurch selektiv unscharf abgebildet und eine Ablenkung vermieden. Bei der Semantischen Tiefenschärfe (Semantic Depth of Field - SDOF) wird nach der Elementrelevanz ausgewählt, welche Objekte im Fokus behalten werden sollen oder wohin die Aufmerksamkeit des Betrachters gelenkt werden soll. Dabei wird die Distanz der Kamera zum Objekt irrelevant. Der Vorteil dieser Methode ist, dass sie intuitiv ist, für Laien einfach verständlich ist.

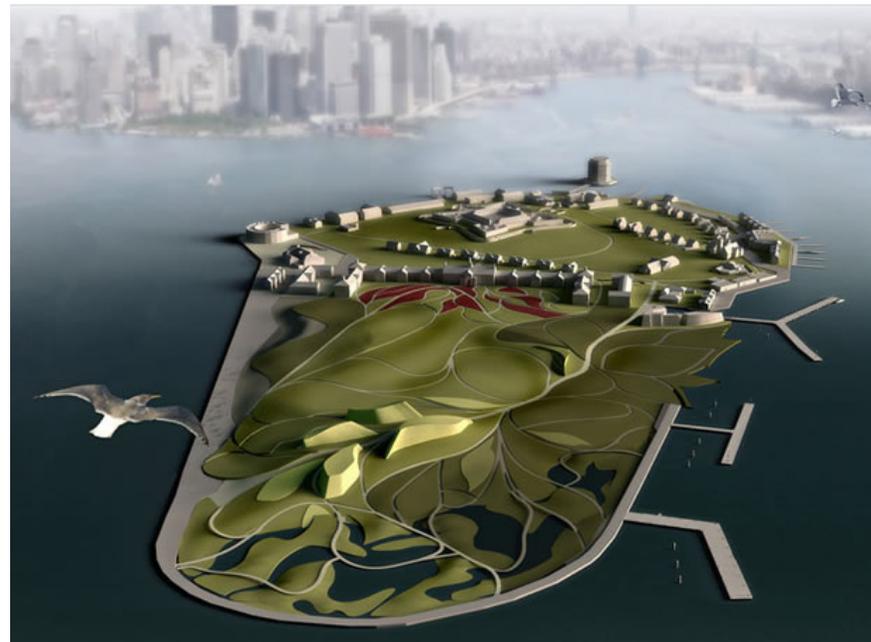


Abb. 28: Einsatz von Semantische Schärfentiefe (Insel bei New York)

(Farb- oder Beleuchtungsmodelle) Fokuselemente können durch Licht- (Abb. 29) oder Farbgebungen (Abb. 30) betont werden. Kontextinformationen werden dabei in Grautönen gezeigt oder wenig beleuchtet. Dabei ist die Verteilung der Helligkeit und Auswahl von geeigneten Farbtönen zu beachten.



Abb. 29: Einsatz von Beleuchtung (Visualisierung WU Wien)

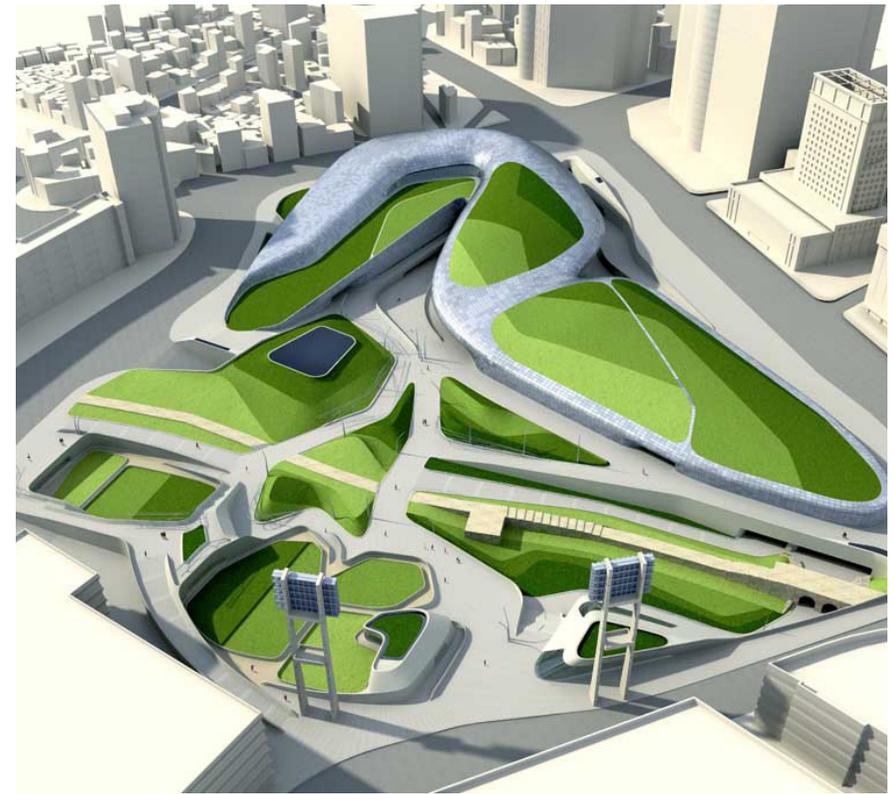


Abb. 30: Einsatz von Farbe (Rendering für das Dongdaemun Design Park)

3. Verwandte Arbeiten

(2D - Darstellungen) Bislang sind expressive Darstellungen in der Architektur hauptsächlich als statische Grafiken verwendet worden. In Abbildung 31 wird eine solche Darstellung gebracht. Hierbei handelt es sich um ein Stadion, in dem die wichtigsten Teile durch eine Explosion hervorgehoben wurden. Ganz unten befinden sich die Eingänge, die Räumlichkeiten für Zuschauer und Spieler, das Spielfeld und die Unterkonstruktion der Tribüne. Zusätzlich ist die Umgebung visualisiert, um das Stadion im Stadtbild verankern zu können. Schaut man genau, so sieht man auch angedeutet die einströmenden Zuschauer in Form von Punkten. Darüber befinden sich in einer explodierten Ansicht die obere Hälfte der Tribüne, die Fassade und die Dachkonstruktion. Die Explosion ist in der Vertikalen angelegt, der Hintergrund nur abstrakt durch einen Farbverlauf angedeutet.

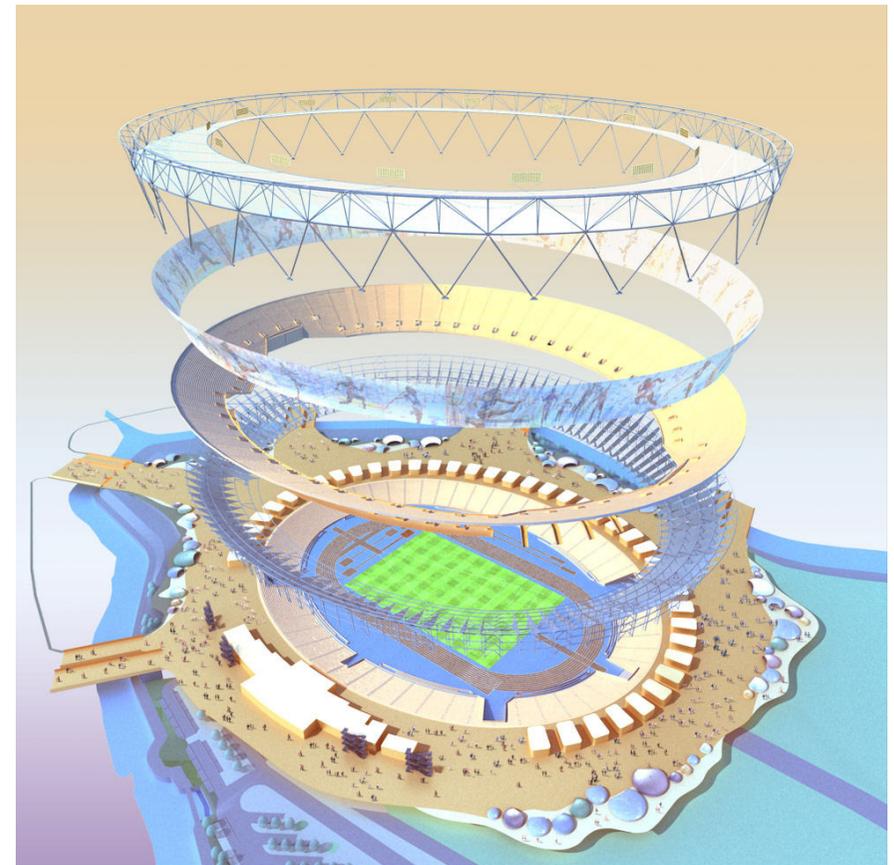


Abb. 31: 2D Darstellung von Exploded View (Olympic Stadium London von Populous)

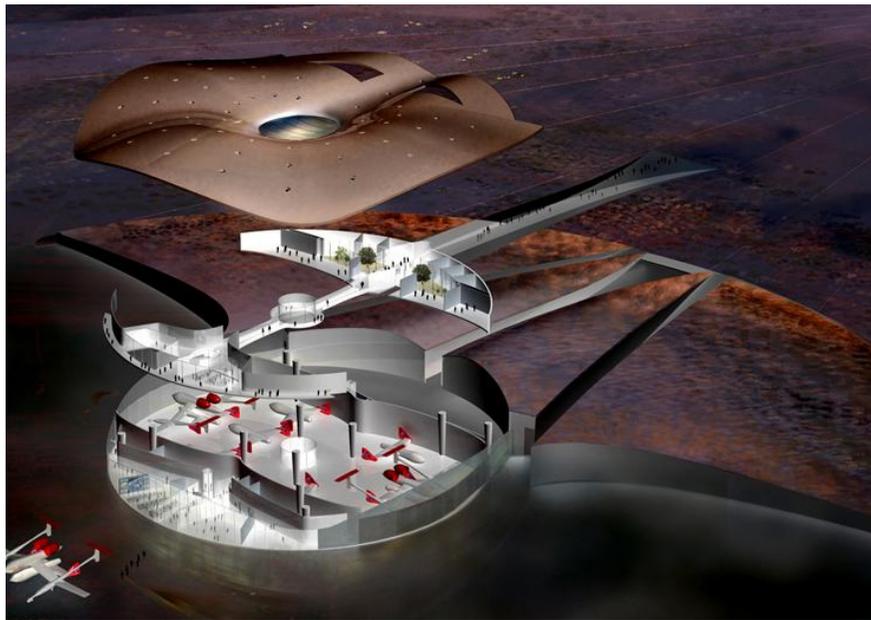


Abb. 32: Spaceport America von Foster + Partners

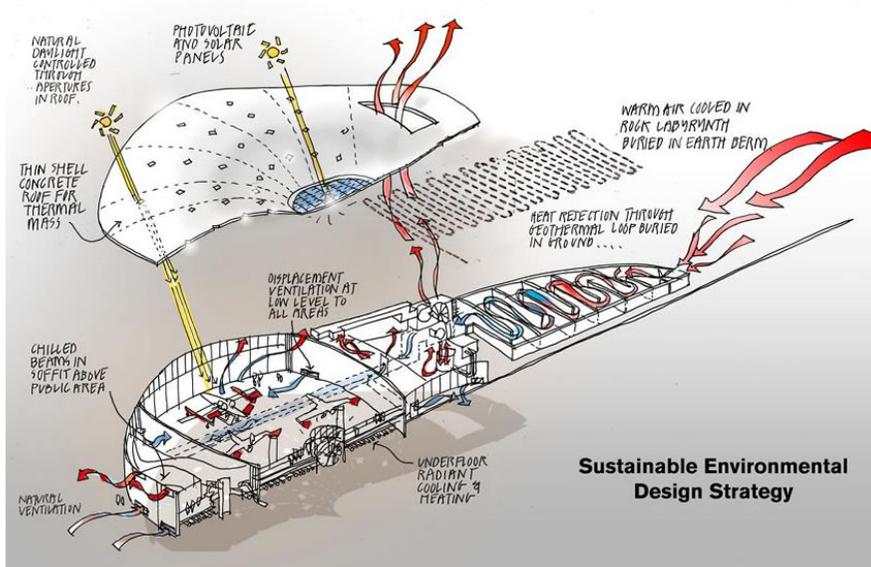


Abb. 33: Spaceport America von Foster + Partners

Weitere Beispiele für die Verwendung solcher Darstellungen in der Architektur finden sich beispielsweise bei der Foster and Partners (Abb.32): Hier wird der Spaceport America als realistisches Rendering dargestellt. Das selbe Projekt wird aber abstrahiert gezeigt (Abb. 33), um bauphysikalische Aspekte darzustellen. Zusätzliche Labels und Guides erläutern dabei die Luftströmung und die Sonneneinstrahlung. Auch sind Techniken wie Cut-Away Views und Ghosted Views schon öfter verwendet worden um komplexe Gebäude übersichtlicher zu präsentieren.

Beim Eiffelturm Wettbewerb zeigten die Architekten Lefevre und Badie ihren komplexen Entwurf der unterirdischen Ebenen durch einen vertikalen Schnitt (Abb.34). Die Schnittebene verläuft mittig durch das gesamte Untergeschoss.

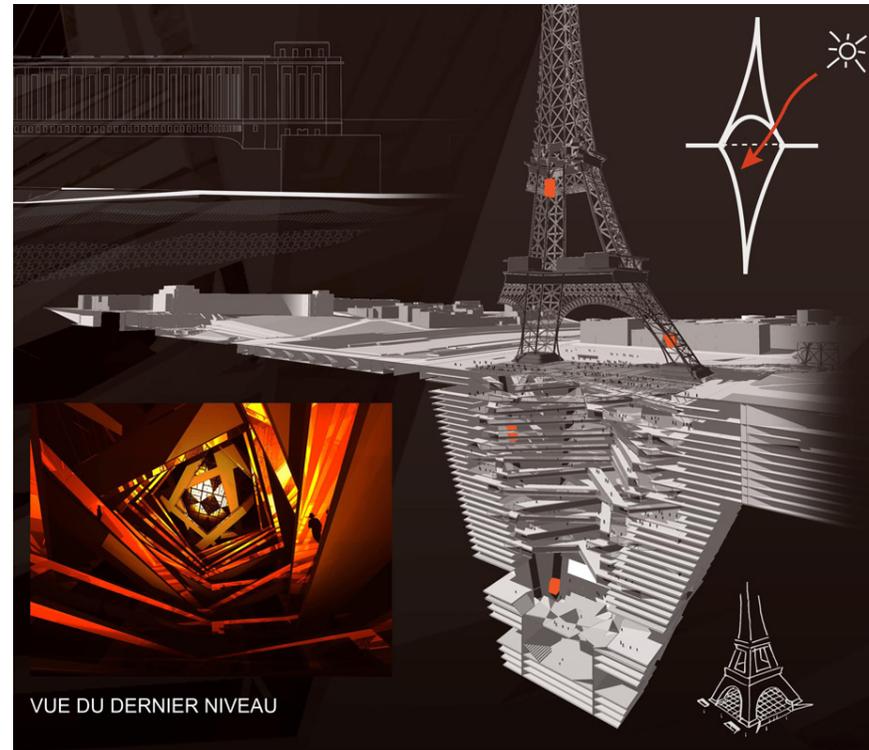
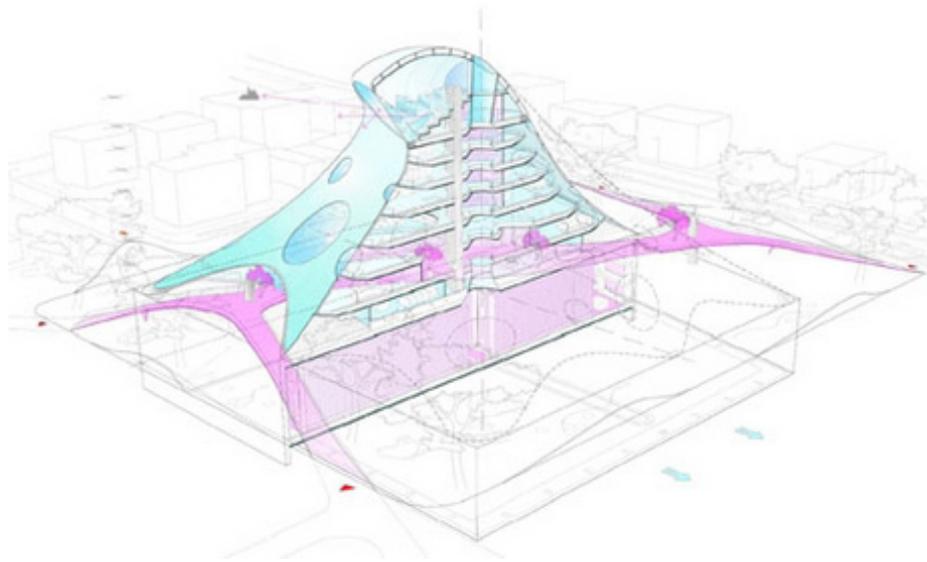


Abb. 34: Wettbewerbsprojekt von Gilles LEFEVRE and Mathieu Badie architects



Eine weitere Technik ist die Abstraktion. Die neue Nationalbibliothek in Prag wurde vom Architekturbüro Future System in stark reduzierter Form und mit der Hilfe von Symbolen dargestellt (Abb.35). Auch wurde die Erschließung und die einzelnen Funktionsbereiche farblich hervorgehoben. Wie im vorigen Beispiel kommen Cut-Aways zum Einsatz. Es wird auch mit Transparenz gearbeitet (Phantom Views). Einige expressive Ansätze, wie zum Beispiel unterschiedliche Abstrahierungstechniken, sind auch in realitätsnahen 2D Visualisierungen zu sehen. Das Ziel ist dabei grundsätzlich, zusätzliche Informationen anzubieten und wichtige Modellteile zu hervorheben.

(3D - Darstellungen) In der Architektur sind interaktive 3D – Explosionen unüblich. Die Echtzeitdarstellungen sind gewöhnlich als Vogelperspektive und Walkthroughs in Benutzung. Konkret konnten wir kein Beispiel für eine solche Darstellungsweise finden, auch wenn die Wissenschaft sich mit der Erzeugung dieser Bilder beschäftigt (siehe Paper „Non-Invasive Interactive Visualization of Dynamic Arch. Environments“ weiter unten).

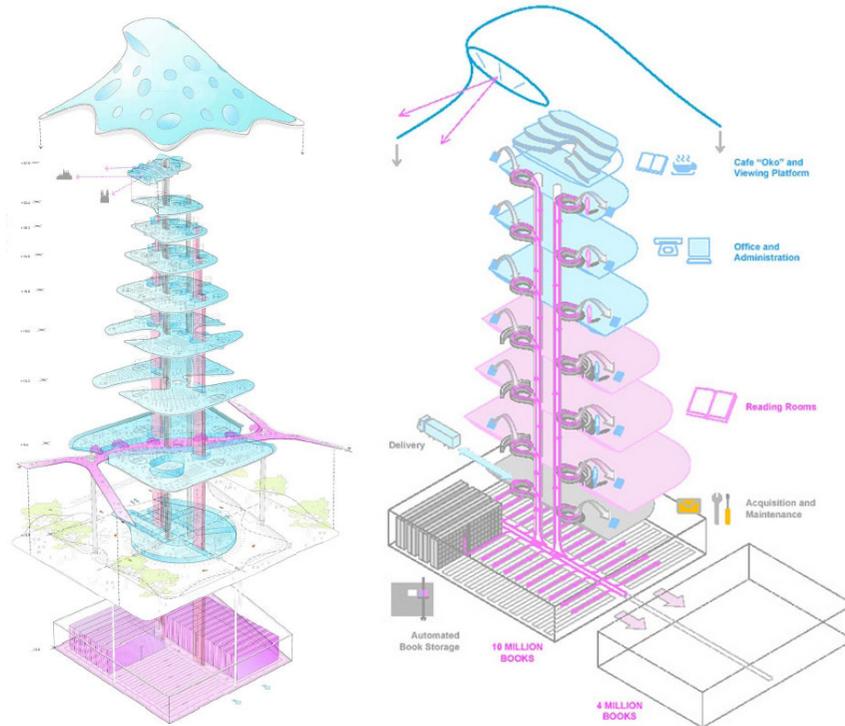


Abb.35: Nationalbibliothek in Prag von Future Systems
a) Cut-Away View; b) Exploded View; c) Exploded View mit Symbole

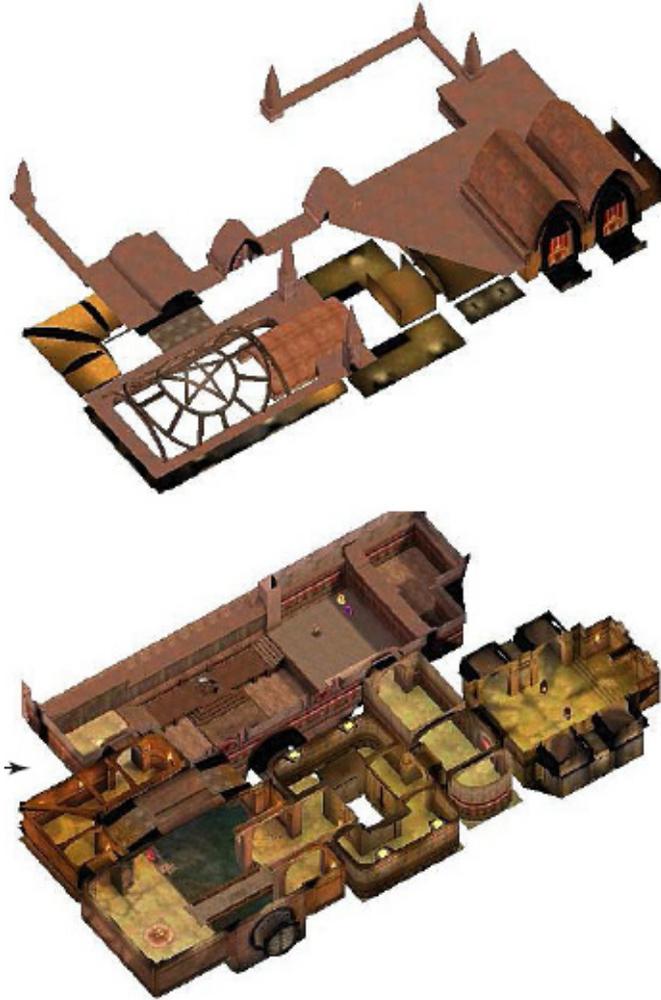


Abb. 36: Automatisch erzeugte Exploded Views von der Spiel Quake III

3.1. Paper: Non-Invasive Interactive Visualization of Dynamic Architectural Environments

Ziel: Automatische Erzeugung von Exploded Views von Spielen

Ansichten: Exploded Views

(Beschreibung) In diesem Paper werden automatisch erzeugte Exploded Views von Gebäuden in Computerspielen präsentiert. Die Exploded Views werden als weitere, wichtige Darstellungsmöglichkeit zu den oft verwendeten „Arcball“ (Rotation und Zoom) und Walk-through Interface beschrieben. Diese bieten zusätzliche Informationen und zeigen die globale Gliederung von architektonischen Räumlichkeiten. Durch die Exploded Views können die Bewegungen und Interaktionen der dynamischen Spieler besser beobachtet werden.

(verwendete Technologie) Es wurde ein System entwickelt, um möglichst automatisiert Exploded Views zu generieren. Die Programmbibliothek Chromium wird dabei angewandt um ein Gebäude in mehrere Ebenen zu segmentieren¹¹. Als Beispiel werden Gebäude des Spieles Quake III vorgestellt (Abb. 36).

(entwickelte Algorithmus) Der erste Schritt, um Exploded View zu kreieren, ist die Analyse der Gebäudegeometrie. Damit werden die Geschossdecken gefunden. Das wird durch Suche der größten Zahl an nach unten gerichteten Polygonen gemacht, was der Findung der Geschoßgrenze entspricht. An dieser gefundenen Grenze werden Schnittebenen platziert und das Gebäudemodell aufgeschnitten. Als zweiter Schritt wird ein Multipass-Renderer verwendet, um alle Geschosse getrennt zu rendern. Die Geschosszahl und die Mindestdistanz zwischen den explodierten Geschossen müssen dabei manuell vom Benutzer eingegeben werden.

(Vergleich zu diese DA) Ähnlich wie dieses Beispiel entwickelt diese Diplomarbeit einen Algorithmus zur automatisierten Gebäudeaufteilung und zum Rendering der Teile. In dieser Diplomarbeit wird dies aber in 3DS Max mittels des MaxScript bewerkstelligt und als Endprodukt werden 2D Grafiken erzeugt. Beim

¹¹ vgl. Niederauer; Houston, 2003

beschriebenen Paper sind die anderen expressiven Darstellungen wie Ghosted Views oder Cut-Aways nicht untersucht worden.

3.2. Paper: Interactive Image-Based Exploded View Diagrams

Ziel: Automatisches Erzeugen von interaktiven Explosionen als 2D Grafiken

Ansichten: Exploded Views

Interaktion: Ein und Aufklappen, Labels

Interaktive Exploded Views erzeugt durch 2D Grafik

(Beschreibung) Der Fokus dieses Papers liegt in der automatischen Erzeugung von Exploded Views für komplexe Objekten. Als Eingangsmaterial werden 2D Grafiken dieser Objekte verwendet und durch das entwickelte System in 2.5D Diagramme mit Interaktivitätsmöglichkeit verwandelt. Dieser grafikbasierte Ansatz hat den Vorteil der Unterstützung beliebiger Renderingstile und eliminiert das Bedürfnis, aufwendige 3D Modelle zu modellieren. Die Resultate dieses Papers demonstrieren, dass einfache 2.5D Diagramm Repräsentationen sehr effektiv sind und viele nützliche Integrationsmöglichkeiten haben. Mit den richtigen Autorenwerkzeugen können solche Diagramme schnell und mit wenig Aufwand erzeugt werden (Abb.37).

(Entwickelte Algorithmus) Dargestellt werden Semiautomatische Werkzeuge für schnelleres Erzeugen von Layern, Explosionsabläufen und Interfaces für eine interaktive Ansicht. Als erster Schritt wird ein 2D Bild mit einem Werkzeug (ähnlich dem magnetischen Lasso von Photoshop) in Teile ausgeschnitten. Diese Objektteile werden danach in die richtige Reihenfolge gestapelt. Die ausgeschnittenen Teile, die immer noch ineinander greifen, werden fragmentiert und richtig angeordnet. Die Achse der Explosion kann vom Benutzer angegeben werden. Am Schluss können Labels, alle Objektteile optional zugefügt werden und durch einen Ankerpunkt platziert werden. Die entwickelte Software für das Anzeigen der dynamischen Exploded Views bietet mehrere Interaktionsmöglichkeiten, wie direkte selektive Explosionsmanipulation, animiertes Expandieren/Zusammenfallen der Teile oder die Suche nach einem

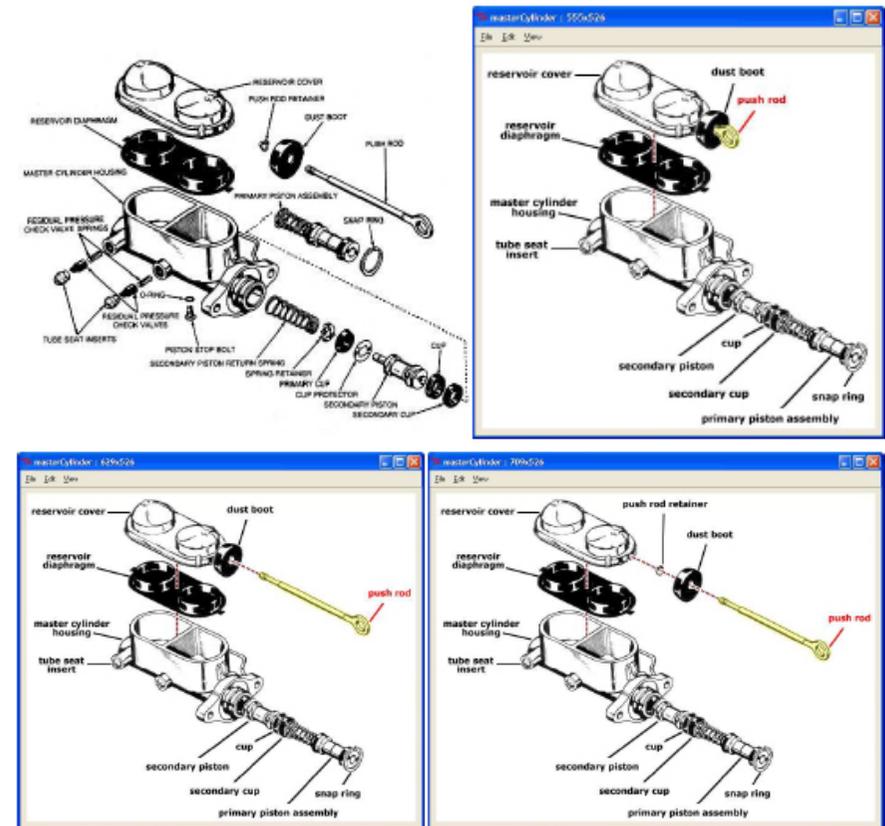


Abb. 37: Automatisch erzeugte Exploded Views von 2D Grafiken

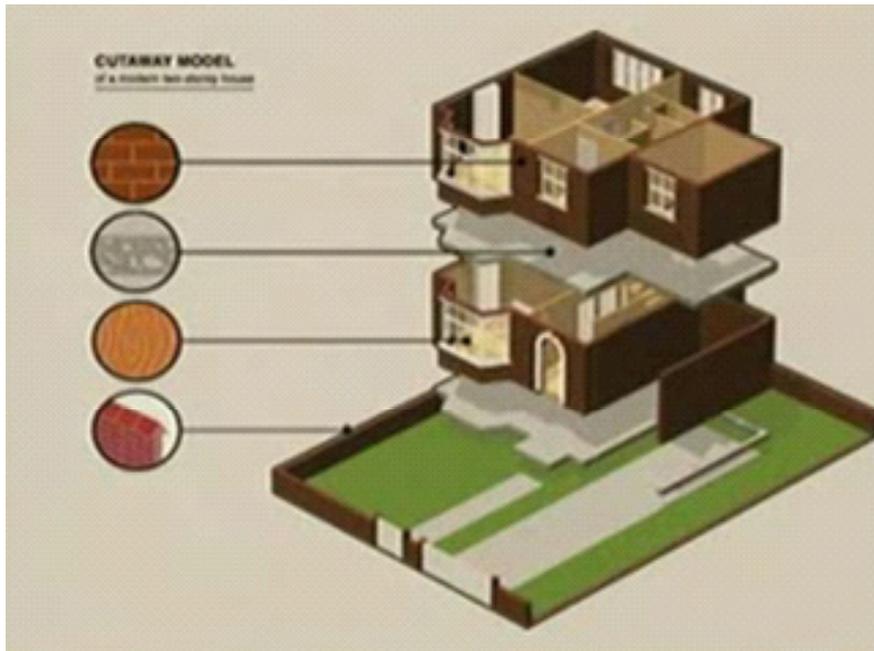


Abb. 38: Music Video „Remind Me“



Abb. 39: Infographik für Areva

bestimmten Teil.

(Vergleich zu diese DA) Das Paper ist der Intention dieser Diplomarbeit ähnlich, im Ansatz hauptsächlich mit 2D Grafiken zu arbeiten um interaktive Visualisierung zu erzeugen. Dieses Paper ist aber nur der Erzeugung der Exploded Views gewidmet und dabei werden die Vorteile anderer Expressiver Visualisierungen (Ghosted Views oder Cut-Aways) nicht berücksichtigt. Um die interaktive Visualisierung anzusehen ist hier eine Software „Viewer“ entwickelt worden. Dieser muss jeder Nutzer auf dem eigenen Rechner installieren. Ein Nachteil ist der fehlende Zoom bei komplexeren Objekten, um Teile detailliert darzustellen. Die Diplomarbeit beschäftigt sich im Unterschied zu diesem Paper ausschließlich mit der Präsentation architektonischer Räume.

3.3. Musikvideo „Remind Me“ und Infografik für Areva von H5

Technik: Animierte 2D Infografiken; Film

Ansichten: Exploded Views, Cut-Aways

(Beschreibung) Das Musikvideo „Remind Me“ (Abb. 38) und das Werbevideo der Firma Areva (Abb. 39) stellen animierte Infografiken von vielen architektonischen Räumlichkeiten dar. Infografiken sind informative Repräsentationen von Objekten, die verbale und visuelle Elemente kombinieren. Bei der Darstellung werden viele Eigenschaften abstrahiert und nur die wesentlichen Dinge beibehalten. In diesen Infografiken sind Exploded Views, Cut-Aways und Ghosted Views integriert, um die Zusammengehörigkeit der Objekte besser zu erklären. Zusätzliche Erklärungen sind als Labels, Guides und Charts eingefügt. In den Architekturdarstellungen werden relevante Informationen mit Zusatzelementen hervorgehoben, wie zum Beispiel das Baumaterial oder die technischen Systeme in einem Gebäude.

(Verwendete Technologie) Als Nachteil dieses Beispiels für Architekturpräsentation könnte die passive Rolle des Betrachters genommen werden, da er im Film nicht aktiv eingreifen kann. Der Ablauf der gezeigten

Ansichten ist vorgeplant und verläuft linear. Die einzige Möglichkeit einer Interaktion und Selbstbestimmung ist das Pausieren und das Rückspulen des Films. Das Ziel dieser zwei Videos ist aber nicht in erster Linie die Architektur zu präsentieren und deswegen wird in diesen konkretem Fall das ausgewählte Medium nicht in Frage gestellt.

(Vergleich zu diese DA) Dieses Beispiel zeigt interessante Möglichkeiten für den Ansatz an Cut-Aways und explodierten Darstellungen in der Architektur. Dabei ist es klar zu sehen, dass diese mehr Informationen als herkömmliche Visualisierungen anbieten. Diese sind aber statisch und das vollständige Gebäude ist nicht zu sehen. Die Diplomarbeit versucht das durch eine interaktive Positionsänderung zu ergänzen.

3.4. Webauftritt von McKinstry

Technik: Flash

Ziel: Webauftritt einer Architektur- und Baufirma

Ansichten: Außenansicht, Exploded Views, Animierte Raumnutzungen

Interaktion: Fassade ausblenden, Geschoss Ein/Auszoomen, Labels Ein/

Ausschalten, Detailinfo für einzelne Geschosse.

(Beschreibung) Die Webseite der amerikanischen Bau- und Architekturfirma McKinstry, erstellt von der Firma Remerinc, ist als ein Gebäude aufgebaut, wobei jedes Geschoss eine andere Gebäudenutzung darstellt. Alle diese Geschosse sind ein wenig auseinander in die vertikale Richtung verschoben, um die Räumlichkeiten besser darzustellen. Jedes Geschoss kann näher herangezoot und in den Vordergrund gezogen werden. Durch Einzoomen wird das Geschoss im Fokus behalten. Alle anderen Geschosse bleiben dann im Hintergrund und sind nur in Grautönen dargestellt (Abb. 40). Alle Räume der Geschosse können durch die fehlende Außenwand und die semitransparente Innenwände ohne Hindernisse betrachtet werden. Es sind noch Möblierungen und animierte Menschen in den Geschossen dargestellt, womit die Raumgröße und Raumnutzung noch klarer zu erleben ist. Alle zusätzlichen Informationen werden in kleine Buttons mit Tooltips aufgebaut.



Abb. 40: Webseite von McKinstry



Abb. 41: T-Com Haus Präsentation

(verwendete Technologie) Diese Webapplikation ist in Flash aufgebaut. Dort sind mehrere Grafiken zusammengesetzt und animiert. Um komplexere Abläufe zu programmieren, steht das Action Script als Skriptsprache zu Verfügung und ist höchstwahrscheinlich für Aufbau einiger Teile verwendet worden.

(Vergleich zu diese DA) Bei diesem Beispiel ist genauso wie in der Diplomarbeit das Internet die ausgesuchte Plattform für eine Präsentation des Gebäudes. Der Zweck dieser Visualisierung ist die Präsentation einer Baufirma, deshalb sind die Grafiken sehr symbolisch und nicht ganz maßstabstreu dargestellt. Die Diplomarbeit hat im Gegenteil die Präsentation der architektonischer Ideen und gebauter Gebäude als Inhalt. Der Darstellungsstil ist deshalb präziser und abstrahierte Elemente sind erst die Möblierung und die Umgebung. Die Präsentation in der Diplomarbeit ist mit Standards aufgebaut, die keine Plugins benötigen und ist optimiert für eine wiederholende Verwendung in anderen Projekten.

3. 5. T-Com Haus

Technik: AJAX + QuickTime Panorama + Flash

Ziel: Gebäude Präsentation

Ansichten: Außenansicht, Geschoss Cut-Away, 360 Panorama

Interaktion: Geschoss Ein/Ausblenden, Panorama und Detailinfo für selektierte Raume

(Beschreibung) Dieses Beispiel ist einen Prototyp für Immobilienpräsentation im Web. Auf der rechten Seite befindet sich ein dreidimensional aufgebautes Haus. Durch kleine Icons können die Geschosse ausgesucht werden und durch Klick die Räume selektiert werden. Wenn ein Raum selektiert wird, können Panoramaaufnahmen dieses Raumes angeschaut und Textbeschreibungen gelesen werden (Abb. 41).

(Verwendete Technologie) Diese virtuelle Gebäudepräsentation ist in einer Ajax-Engine in Kombination mit Flash, XHTML/CSS-Layout und Apple Quicktime realisiert. Durch den Einsatz von AJAX wurde ein Nachladen der Inhalte ohne

Wartezeiten ermöglicht. Durch die Verwendung von diesen Technologien ist auch ein Zugang für die Benutzer gegeben, die die Quicktime- und Flash-Elemente nicht laden können.

(Vergleich zu dieser DA) Der Präsentationsablauf erfolgt ganz anders als in dieser Diplomarbeit. Die Lokation der Räume im Haus ist von der Zeichnung gut abzulesen, aber die Orientierung der gesehenen Räume könnte trotzdem missverstanden werden. Diese zwei Elemente sind in der Diplomarbeit zusammengefügt und die Probleme der Verdeckung durch expressive Darstellungen gelöst. Sowohl der Diplomarbeit wie auch in diesem Ansatz ist gemein, dass auf Standardtechnologien für die Implementierung der Anwendung zurückgegriffen wird.

4. Entwicklung einer semi-automatischen, expressiven Architekturvisualisierung

(Ergebnis) Ein Prototyp wurde erstellt, um expressiven Darstellungen (Exploded Views, Cut-Aways, Ghosted Views usw.) auf Gebäudeebene zu erforschen. Dazu wurde ein Modell von TownTown im dritten Wiener Gemeindebezirk mit verschiedenen Darstellungsformen aufgebaut, um das Innere des Gebäudes, die Prozesse und die technischen Prinzipien besser zu zeigen. Die wichtigsten Bauteile wurden zudem mit Zusatzinformationen versehen, die durch Anklicken des Modells am Bildschirm angezeigt werden können. Ein grafisches Interface mit zusätzlichen Kontrollbuttons wurde entworfen, um z.B. die Darstellung von Modellgruppen regeln zu können.

Die Visualisierung ist eine (2.5D) axonometrische Explosionsgrafik mit zusätzlichem Einsatz von Ghosting, Labels und Guides. Im Vergleich zu einem 3D-Modell ist dieses 2.5D System speicherschonend und kann ohne besondere Zeitverzögerung auch bei langsameren Internetverbindungen dargestellt werden. Ein weiterer wichtiger Grund für die Verwendung von 2D Grafiken für dieses Projekt war auch die Möglichkeit, beliebige Darstellungsstile ohne großen Aufwand zu verwenden. Die Grafiken sind einfach zu ersetzen und die interaktive Präsentation kann abhängig vom Projekt als abstraktes Diagramm oder als realitätsnahe Visualisierung verwendet werden.

4.1. Anwendungsbeispiel: Visualisierung von Town Town

Allgemeine Eckdaten zum Zielgebiet

(Lage) Town Town befindet sich in den 3. Wiener Gemeindebezirk bei der U-Bahnstation Erdberg. Der Standort ist begrenzt durch die Erdbergstraße, Wünzlerstraße, die Südosttangente A23 und die Ost-Autobahn A4 (siehe Abb.42).



Abb. 42: Lageplan von TownTown

Auf diesem Areal befand sich früher ein Wagenabstellplatz der U-Bahn. Mit der Idee, dieses Areal nezugestalten und zu verwerten, hat dann im Jahr 2000 das Bauprojekt Town Town angefangen, welches sich derzeit in der dritten Bauphase befindet (Fertigstellung 2011 geplant). Die Umsetzung leitet das Architekturbüro Baumschlager und Eberle. Das „Landmark“ Gebäude (Gebäude 21) soll jedoch vom Architekturbüro Coop Himmelb(l)au gebaut werden (Abb.43).

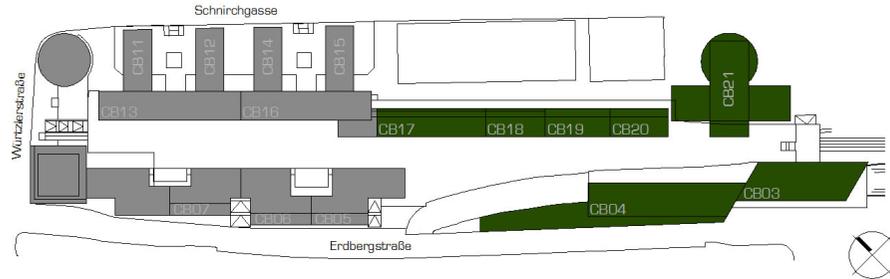


Abb. 43: Lageplan von TownTown

(Architektonisches Konzept) Durch dieses Bauprojekt wird ein Büro- und Geschäftsgebiet in Form einer Überbauung entwickelt. Das Gebiet beinhaltet eine Piazza, eine Tiefgarage und 21 Gebäude. Die Piazza ist eine großzügige Fußgängerzone und dient als Erschließungsbereich für alle 21 umgebende Gebäude. Im Bereich der U3-Station sind zwei Hochhäuser geplant, eines der Gebäude trägt ein für die Öffentlichkeit zugängliches Aussichtsgeschoss.

(Nachhaltigkeit und Bauphysikalische Konzept) Bei der Entwicklung von Town Town wird ein großer Wert auf Umwelt und Ökologie gelegt. In Hinsicht auf Klimälösungen werden in allen Gebäude innovative Systeme, wie beispielsweise Fernkälte und Betonkernaktivierung, eingesetzt. Die Fernkälte nutzt vorhandene Wärme aus Müllverbrennungsanlagen, Industriebetrieben oder aus der Stromproduktion als Primärenergie. Neben wesentlich geringerem Energiebedarf geben diese Gebäude keine Treibhausgase an die Umwelt ab, zudem sind sie auch in der Wartung sehr kostengünstig¹². Bei der Betonkernaktivierung (oder Betonkerntemperierung) wird die Fähigkeit der Decken, thermische Energie zu speichern, genutzt. Durch diese Methode werden die Räume kostengünstig

vgl. www.wienenergie.at, 2009

geheizt. TownTown hat das Gold-Gütesiegel im Rahmen der internationaler Kongress- und Fachausstellung für nachhaltiges Bauen erlangt.

Beschreibung der Visualisierung

Die ausgearbeitete Visualisierung für den in Abb. 42 angedeuteten Teil des Projekts TownTown ist in drei Bereiche geteilt: Innen- und Außenraumvisualisierung (Abb. 44a), Wegführung (Abb. 44b) und Visualisierung des bauphysikalischen Systems.

In der Innen- und Außenraum Visualisierung sind die Raumprogramme der Gebäude 3, 4, 21 und 17-20 dargestellt. Dabei ist die Außenansicht der Gebäude in ihrer Umgebung dargestellt. Die ausgewählten Geschosse der Gebäude können vertikal verschoben werden. Bei Regelgeschossen ist nur ein Geschoss ausgearbeitet und gezeigt. Die Innenräume können eingezoomt werden und sind durch zusätzlich Informationen in Form von Fotos und Texten angereichert. Die zweite Darstellungsform ist schematisch und soll ein Wegweiser durch die ganze Anlage bieten. Die Fassaden der Gebäude ist abstrahiert und transparent dargestellt. Die Innenräume sind zweidimensional, wobei unterschiedliche Nutzungen durch eigene Farben gekennzeichnet sind. Die Erschließungselemente sind dreidimensional ausgearbeitet und durch kleine Symbole zusätzlich hervorgehoben. Alle Bereiche sind mit Labels beschriftet worden.

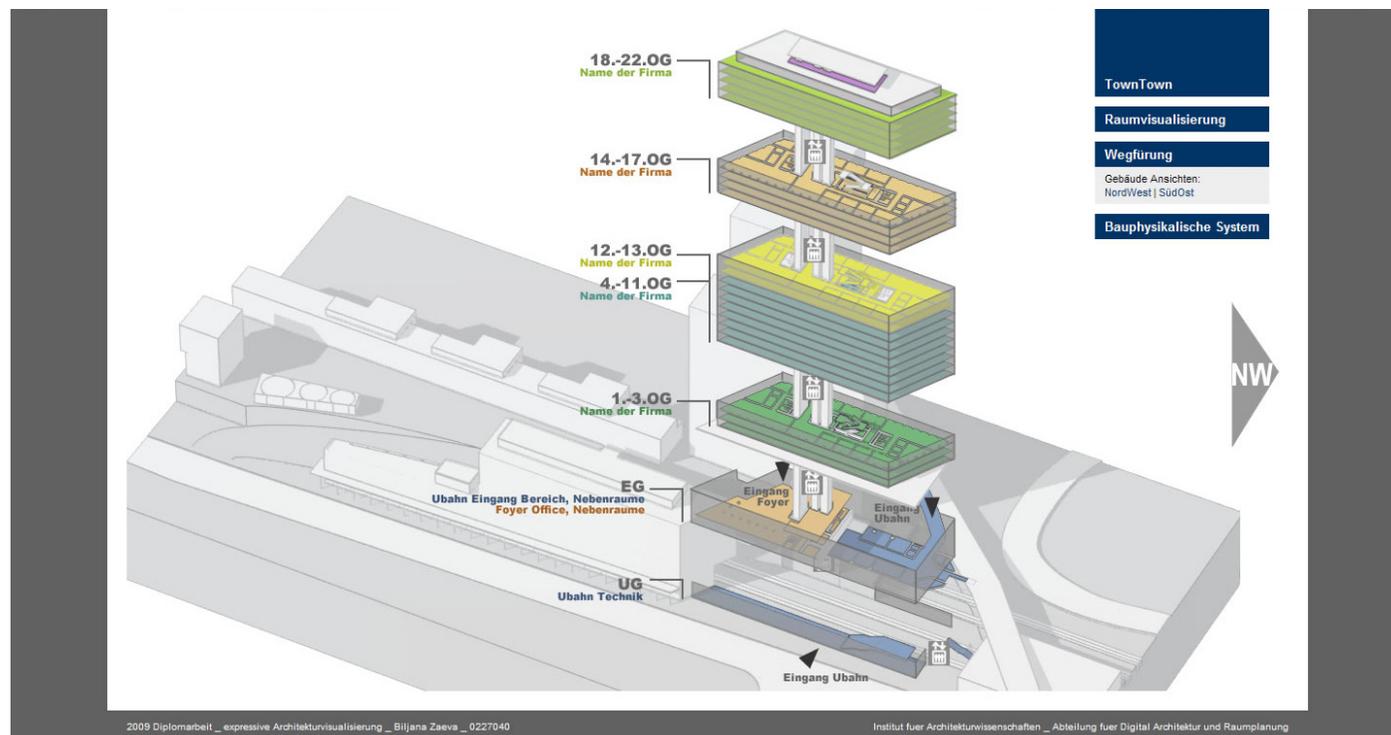
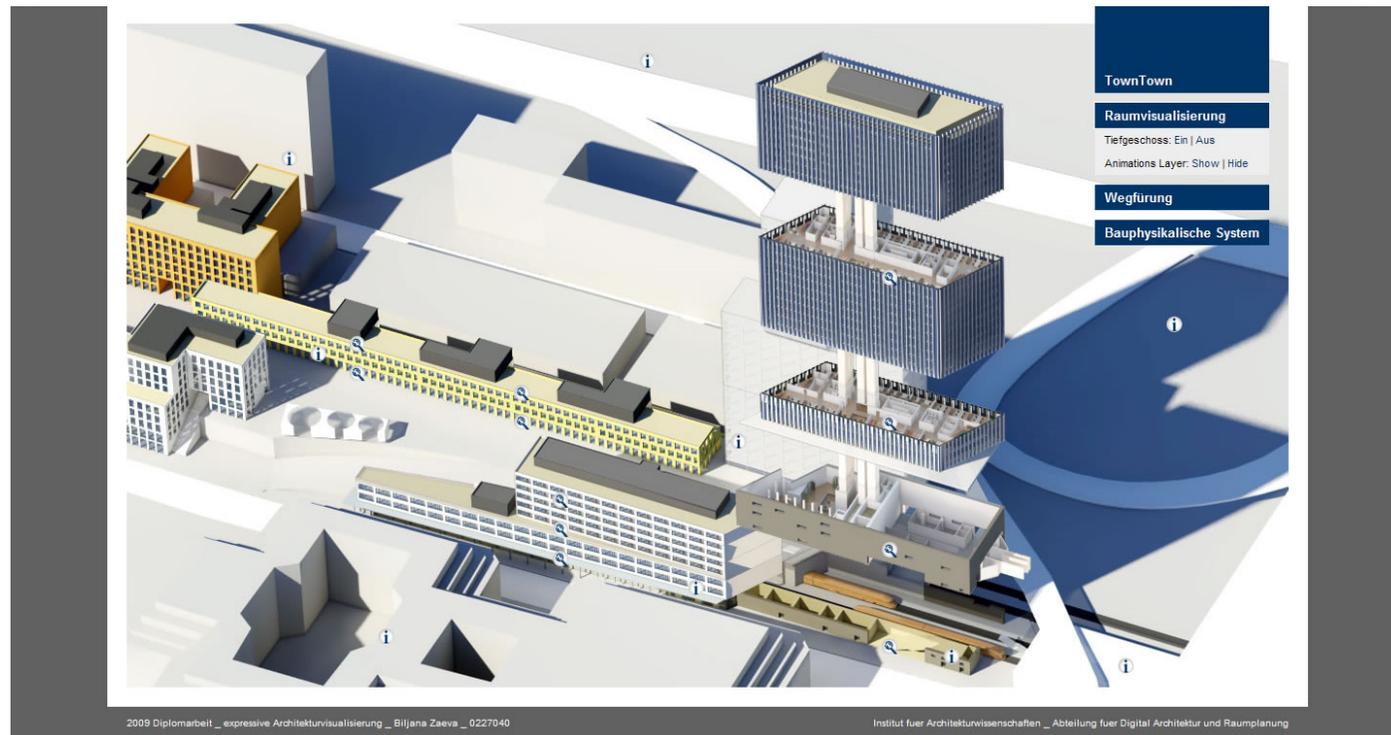


Abb. 44: a) Innen- und Außenraum Visualisierung, b) Wegführung

Workflow für die Implementierung

Das Diagramm in den Abbildungen 45 und 46 beschreibt den benutzten Workflow für die Erzeugung der expressiven Webvisualisierung. Die gesamte Arbeit würde in fünf Abschnitte aufgeteilt: Vorbereitung, Modellierung, Aufbereitung des Rendering, Erstellung expressiver 2D Teilgrafiken und Erstellung der Webpräsentation.

(Pläne, Fotos, Texte zusammenstellen und aufbereiten) Zuerst war es wichtig zu entscheiden, welche Medien für Umsetzung des Projekts benötigt werden. Dies waren hauptsächlich Pläne, Fotos, Texte, Texturen und 3D Modelle. Pläne (Grundrisse, Schnitte und Ansichten) wurden im DWG Format von der Firma Soravia, welche die Projektentwicklung für TownTown durchgeführt hat, zu Verfügung gestellt, und konnten anschliessend in AutoCAD für die Visualisierung bereinigt und vereinfacht werden. Unnötige Layer und Schraffuren wurden dabei gelöscht und unterbrochene Linien verbunden. Dies war notwendig für die spätere Extrusion in 3DS Max. Die Gebäude, die schon errichtet waren, wurden fotografiert, um Materialien korrekt visualisieren zu können. Texturen für die Gebäudeteile stammen zum Teil auch von Online- oder CD Bibliotheken, und sind in Photoshop nachbearbeitet worden. Fertige 3D Elemente für Ausstattung der Innenräume sind auch von Internetbibliotheken extrahiert worden. Projektbeschreibungen wurden aus der vorhandenen Dokumentation zusammengestellt und werden auch in der Visualisierung verwendet.

(Modellieren) Alle Gebäude des Projekts wurden in 3DS Max neu modelliert. Dabei war das LOD (Level of Detail) an eine Internetpräsentation angepasst, sprich: es wurden wenig Polygone verwendet. Bei diesem Projekt wurde auch vorgeplant, welche Innenräume für die Präsentation relevant sein könnten und dementsprechend genauer ausgearbeitet und inszeniert werden sollten. Die Schlussendliche Ausarbeitung ist in den Detailgrafiken zu den Geschossen sichtbar (Abb 47).

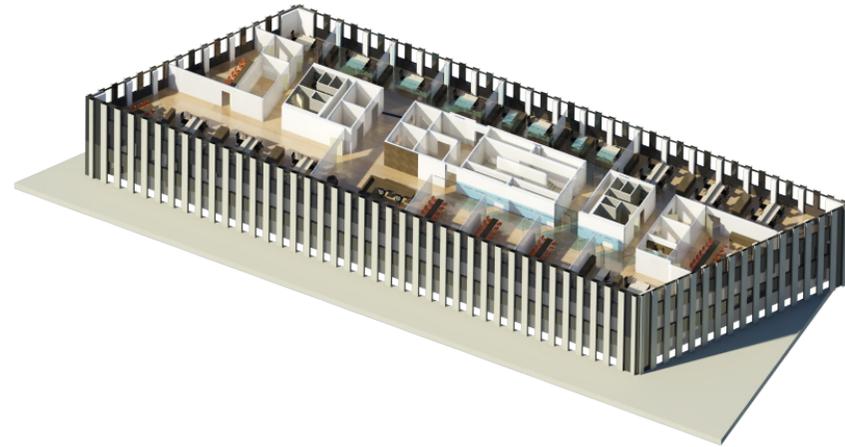


Abb. 47: Regelgeschoss

(Materialvergabe und Beleuchtung) Als Rendering Engine für realistische Darstellung wurde Mental Ray verwendet, das schon in 3DS Max integriert ist. Dabei wurden zuerst die Lichtquellen (Sonnen- und Himmelslicht) platziert und eingestellt. Für die schematische Darstellung wurde das Ink 'n Paint Material von Mental Ray verwendet, welches eine Oberfläche in künstlerischer Darstellungsform erzeugt.

(Rendering Plug-In) Mit Hilfe eines automatischen Plugins können zurzeit alle auf dem beschriebenen Weg vorbereiteten Gebäudeteile automatisch gerendert und gespeichert werden¹³. Dazu wurde eine Kamera mit orthographischer Darstellung platziert und auf einzelne Ausschnitte der Szene ausgerichtet. Alle Objektgruppen wurden sukzessiv mit dem Plugin selektiert in unterschiedlichen Auflösungen gerendert. Für die 360° Rotation wurde ein Kreis als Pfad der Kamera eingefügt. Als Fokus dieser Kamera und Zentrum des Kreises wurde der Schwerpunkt des Gebäudeteils ausgewählt.

(Nachbearbeitung) Die fertigen Grafiken der Gebäude wurden danach in Photoshop bearbeitet und um zusätzliche Transparenz erweitert (etwa für Ghosted View). Weiters wurden Frames der Animationen als animierte GIFs zusammengefügt. Im selben Programm wurden Linien als Guides eingefügt, um

¹³ das Skript könnte auch erweitert werden, um das Modell auch in Geschosse aufzuteilen

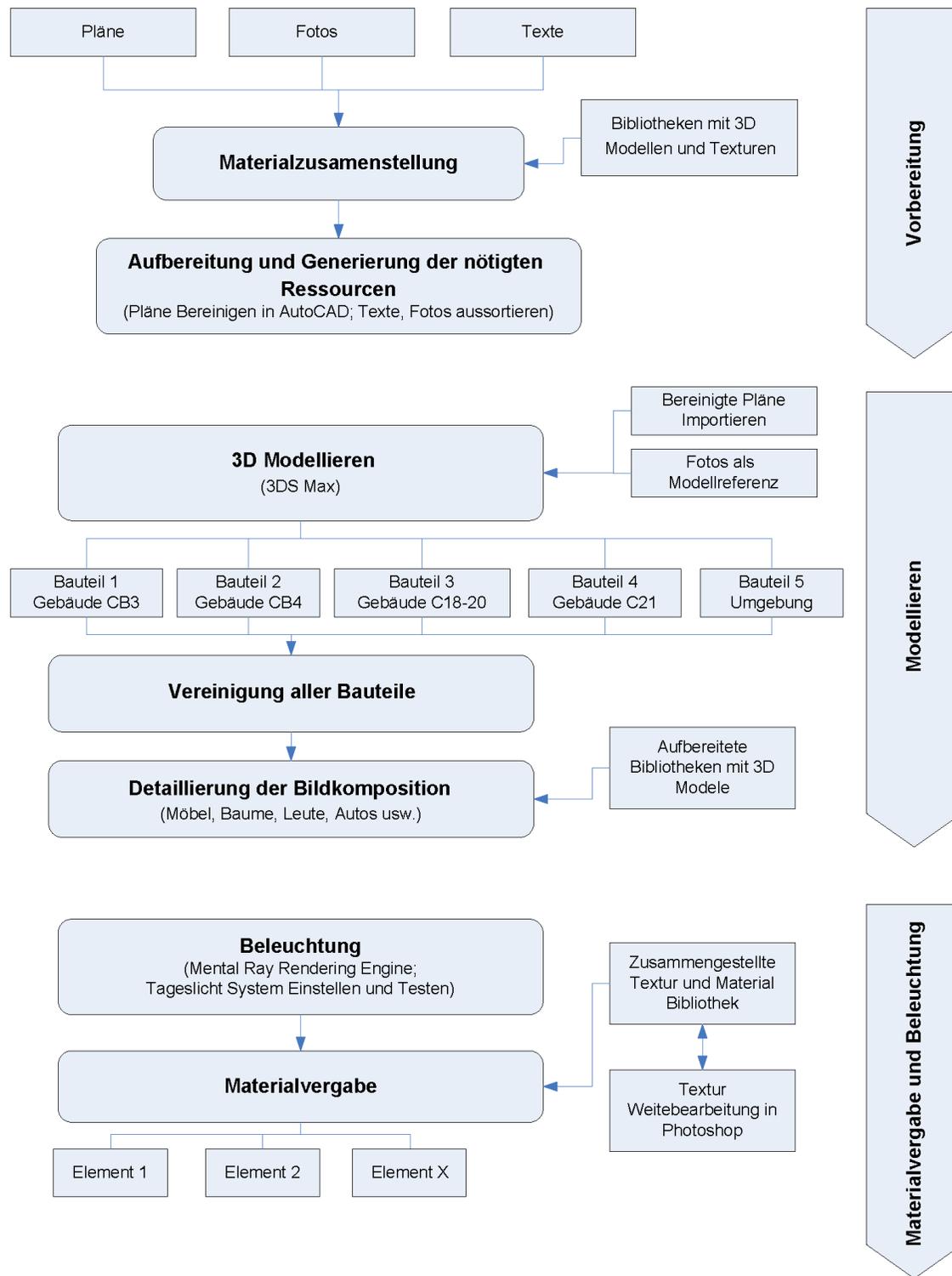


Abb. 45: Workflowdiagramm (Abschnitt: 1-3)

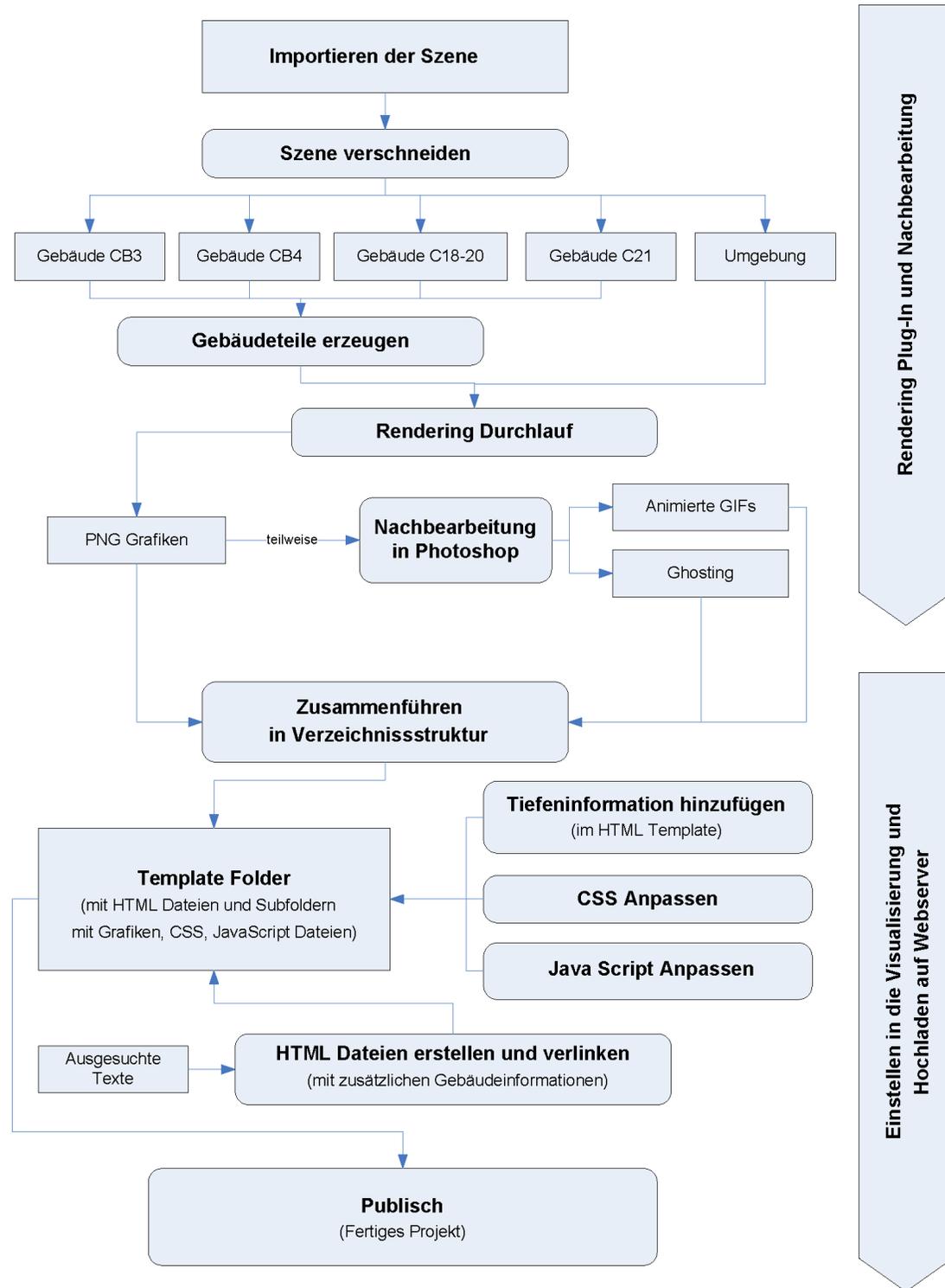


Abb. 46: Workflowdiagramm (Abschnitt: 4-5)

die Explosionsrichtung klarer zu zeigen. Weiter sind Lifte eingefügt worden.

(Einstellen in die Visualisierung und Hochladen auf Webserver) In einem nächsten Schritt wurden die Grafiken in die vorgegebenen Layer des Prototyps gebracht und verlinkt. Dafür waren Grundkenntnisse von HTML und CSS notwendig. Für das Beispiel Town Town werden drei Layer gezeigt: die Visualisierung der Gebäudeteile, eine diagrammatische Ansicht des Gebäudes und das bauphysikalischen System. Rein technisch können aber beliebig viele Layer und Gruppen formiert werden. Die beschreibenden Texte und Bilder wurden in eigenen HTML Dateien editiert und anschließend mittels Icons zur Visualisierung verlinkt. Eine nachträgliche Änderung der Meta-Informationen zum Gebäude ist mit diesem Ansatz rasch möglich. Die Applikation wurde mit allen Grafiken, Skripts, HTML- und CSS-Dateien auf einen Webserver geladen.

Technische Implementierung

3DS Max wurde für das Rendering und die Weiterverarbeitung ausgewählt. Bei dieser Auswahl wurde darauf geachtet, dass das Programm folgende Anforderungen erfüllt:

- Import von externen modellierten Bauobjekten soll leicht möglich sein.
- Unterstützung verschiedener Renderingstile.
- Gute Arbeitsgeschwindigkeit im Umgang mit größeren Dateien.

Die objektorientierte Skriptsprache MAXScript innerhalb von 3ds Max diente als Schnittstelle für die individuelle Anpassung des Programms in Form eines Plugins. Dieses fasst die Arbeitsschritte für die Erzeugung der Renderings von verschiedenen Gebäudeteilen zusammen (Abb. 48). Nach Durchlauf des Plugins liegen 2D-Grafiken in einem definierten Verzeichnis vor, die dann von der Visualisierung geladen und angezeigt werden können.

(Verwendete Grafiken) Es sind sowohl animierte als auch statische 2D Grafiken mit transparentem Hintergrund zum Einsatz gekommen. Diese wurden übereinander gelagert dargestellt (Multilayer System mit einer Tiefe pro Bild),

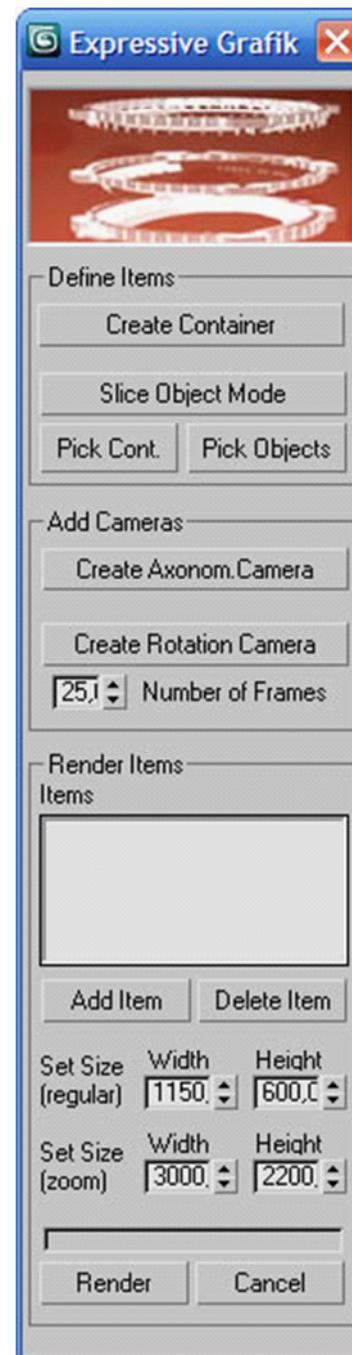


Abb. 48: PlugIn für 3DS Max

wobei einzelne Layer ein- oder ausgeschaltet sowie verschoben werden können. Auf Wunsch der Betrachter können zudem weitere Informationen zu jedem Gebäudeteil angezeigt oder verborgen werden, diese sind als Symbole über allen anderen Grafiken positioniert. Von den Dateiformaten wurden PNGs und animierte GIFs verwendet. Für Bilder mit vielen Farben oder Farbverläufen wäre das JPG Format zwar speicherschonender, es hat aber den Nachteil, dass ein Alphakanal nicht vorhanden ist. Eine Abhandlung und ein Vergleich zwischen den verschiedenen Grafikformaten für das Web findet sich im Anhang (ab Seite 63). Um eine einwandfreie Verwendung der Präsentation bei unterschiedlichen Internetverbindungen zu ermöglichen, wurde stark auf die Dateigrößen der Grafiken geachtet. Durch Bildbearbeitungsprogramme wie beispielsweise Photoshop können die Bilder optimiert werden um diese zu reduzieren. Dabei musste teilweise zwischen Qualität und beanspruchten Speicherplatz ein Kompromiss gemacht werden. Das Bildbearbeitungsprogramm bietet in der Regel einen Vorschaumodus, wo die Qualität vor dem Speichern kontrolliert werden kann.

(Benutzung von Standards und Bibliotheken für die Visualisierung) Vorerst wurden im Zuge der Recherche die technischen Möglichkeiten der Erstellung der Interaktion erkundet und anschließend eine für dieses Projekt passende Technik ausgewählt. Dabei wurde darauf geachtet, genormte Internetstandards zu verwenden. Den Aufbau der Internet Präsentation wurde ausschließlich in HTML und CSS realisiert, für die Interaktion wurde Java Script genommen. Dabei kam eine Open Source Java Script Bibliothek (jQuery) zum Einsatz, die eine Vielzahl von Funktionen für die verschiedenen Aspekte der Webprogrammierung mitbringt. Zudem hat diese Bibliothek eine große Zahl an Entwicklern und damit auch eine breite Benutzerbasis für etwaigen Support in der Entwicklungsphase.

(Testung auf mehreren Browsern) Der Begriff Cross-Browser Kompatibilität bezeichnet die Fähigkeit von Webseiten, unabhängig vom jeweiligen Browsertyp die gleiche Ausgabe zu erzeugen. Während der Entwicklung wurde auf diese Eigenschaft größten Wert gelegt, wobei die Konzentration auf Standards einen erheblichen Beitrag zum Funktionieren in allen modernen Browsern leistete. Dazu müssen HTML und CSS Code auf Richtigkeit geprüft werden. Browser

sind oft tolerant für Fehler in der Programmierung, verhalten sich aber in der Folge nicht wie gewünscht. Es gibt daher unterschiedliche Werkzeuge, um den Code vorab zu kontrollieren. Für die vorgestellte Webpräsentation wurden die Gültigkeitsprüfungen des W3 Consortium verwendet¹⁴. Verschiedene Browser können den JavaScript Code verschieden interpretieren und stellen den Inhalt unter Umständen auch anders dar. Dabei handelt es sich nicht automatisch um Fehler des Browsers, sondern oft um unterschiedliche Standardwerte die verwendet werden. Deshalb wurde ein Test mit Mozilla, IE(6, 7, 8), Opera und Chrome durchgeführt. Für das Testen an den älteren Versionen des IE wurde auf einem virtuellen Computer (Virtual Machine) durchgeführt. Bei diesen Tests wurde zum Beispiel entdeckt, dass der IE6 nicht mit dem transparenten Hintergründen der PNG Dateien umgehen kann. Dieses Manko wurde durch ein zusätzliches Skript gelöst.

(Testung mit unterschiedlichen Bildschirmgrößen) Bei einem Webauftritt muss auch berücksichtigt werden, dass Benutzer mit unterschiedlichen Bildschirmgrößen die Präsentation abrufen. Ein gleichzeitiges horizontales und vertikales Scrollen sollte dabei vermieden werden. In der Zeit des Schreibens diese Diplomarbeit wurden die meisten Webauftritte für eine Bildschirmauflösung von 1024x768 Pixel optimiert, weil nur eine sehr geringe Anzahl an Benutzern kleinere Bildschirme verwendet. Wenn ein Webauftritt auf einer kleinen Bildschirmgröße entwickelt wird, sollten einige zusätzliche Faktoren berücksichtigt werden. Webbrowser werden zum Beispiel nicht immer mit der vollen Größe geöffnet oder die Lesezeichen sind neben den Webinhalten sichtbar. Dies bedeutet eine kleinere Bildschirmfläche für die Applikation selber.

(Testung mit unterschiedlichen Textgrößen) Textgrößen, die durch CSS oder HTML Code definiert sind, können oft durch Standardwerte unterschiedlicher Browser oder Betriebssysteme überschrieben werden. Um das zu vermeiden, wurden alle Formatierungen bei der Entwicklung der vorliegenden Webanwendung ausschließlich in Pixel definiert. Diese Vorgangsweise könnte

¹⁴ <http://validator.w3.org/> (für HTML Kontrolle)

<http://jigsaw.w3.org/css-validator/> (für CSS Kontrolle)

für Leute mit schlechterer Sicht nachteilig sein, sie ist jedoch leicht änderbar (beispielsweise Anpassung für bestimmtes Zielpublikum).

Näheren Beschreibung des Plugins

Als ersten Schritt wird die vorliegende Szene in zusammengehörige Teile gruppiert (Abb. 49), so dass Gebäude nun isoliert für die weitere Verarbeitung vorliegen. Dann wird jedes Gebäude (manuell) in Geschosse aufgeteilt. Bei Gebäuden mit Regelgeschossen ist es sinnvoll, ein Geschoss nur ein Mal zu zeigen. In allen anderen Fällen wird für jedes Geschoss eine eigene Gruppe angelegt.

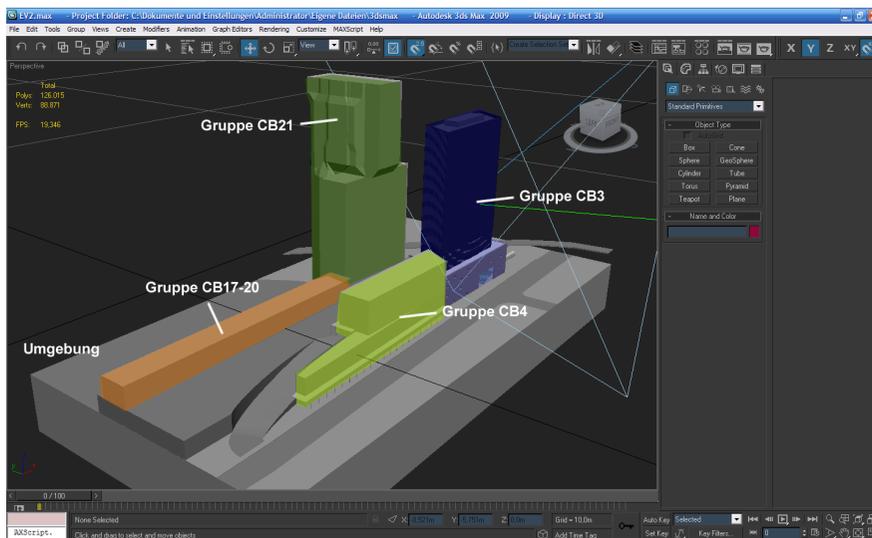


Abb. 49: Definition von Elementengruppen

Der Algorithmus dazu:

- Schneidegrenzen in Gebäude legen (z.B. lt. Paper „Non-Invasive Interactive Visualisation of dynamic Arch.Enviroments“ aus Siehe Kapitel 5.5). Die Schneidegrenzen können dabei entweder als Körper oder als Ebenen definiert werden.
- Gebäude mit Schneidgrenzen verschneiden (Max Werkzeuge: Compound Objects/ProBoolean; Compound Objects/ProCutter, siehe Abb.50). Alle Objekte, die in einer Verschneidung erzeugt wurden, gruppieren und in ein neues Objekt umwandeln (Max Werkzeuge: Menu Group; Create Selection Set).

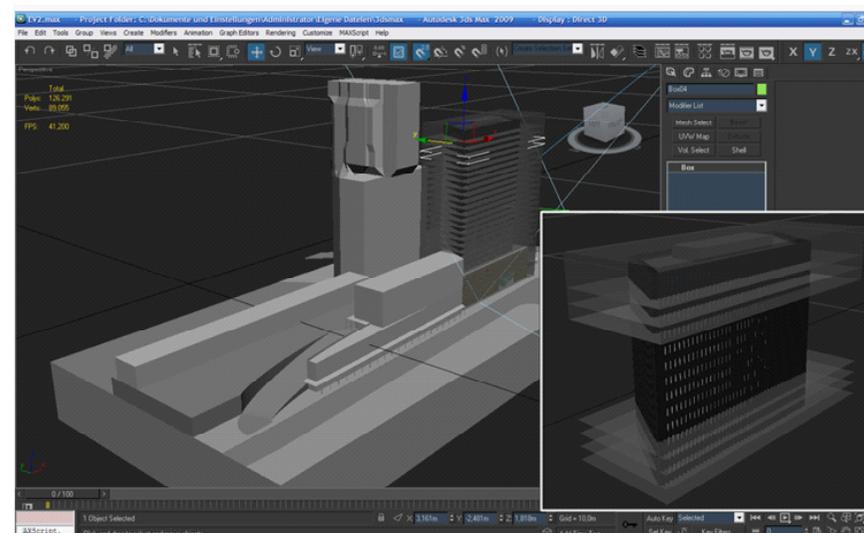


Abb. 50: Definition von Gebäudeteile

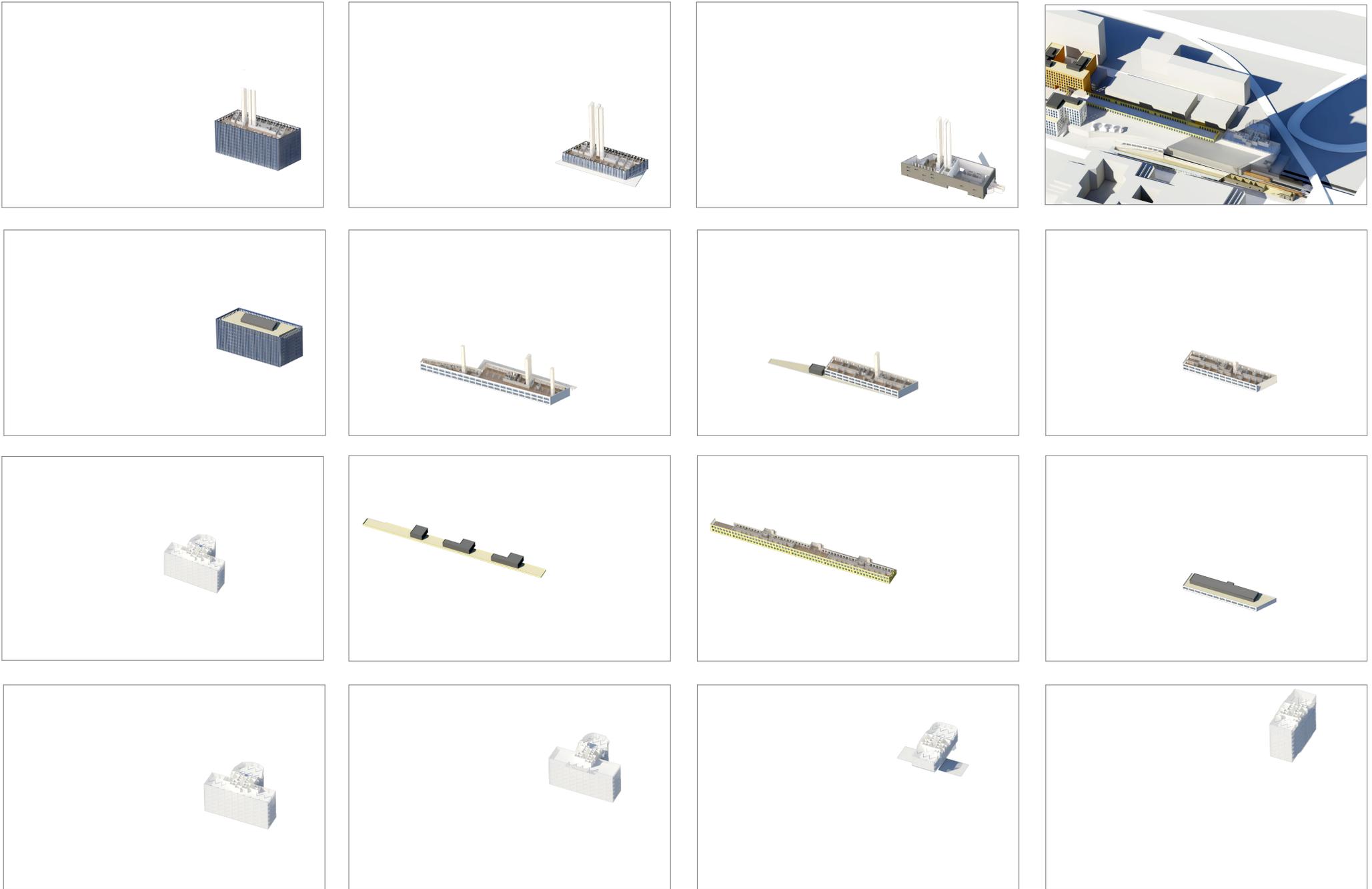


Abb. 51: Erzeugte Grafiken von den Rendering Durchlauf

Als nächstes wird die Position der Kamera definiert. Es wird mit einer orthografischen Perspektive gearbeitet, weil bei einer perspektivischen Visualisierung Verzerrungen entstehen würden. Die Seite, von der das Gebäude am besten zu sehen ist, muss in diesem Schritt ausgesucht werden. Bei Modellen, in denen alle Seiten relevant sind, können mehrere Kameras platziert werden (zum Beispiel: Nord-West und Süd-Ost Kamera). Dabei sind einige Regeln zu beachten, um das bestmögliche Ergebnis von der explodierten Darstellung erhalten zu können:

- Die Kamera soll immer höher als das Modell platziert werden.
- Der Zielpunkt der Kamera soll im Zentrum des Gebäudes positioniert werden.
- Um gute Sichtverhältnis für den Innenraum und den Außenraum zu gewährleisten, ist der Winkel zwischen dem Zielpunkt und der Position der Kamera relevant. Bei kleineren Gebäuden wird ein Winkel von 20-30° und bei höheren ein Winkel von 50-60° als geeignet empfunden.

Wenn alle Teilelemente in der Szene definiert sind, kann mit dem Renderingdurchlauf angefangen werden. Das geschriebene Plugin erzeugt dabei automatisch alle Gebäudeteile als Grafik. Technisch gesehen wird jeweils nur ein Gebäudeteil für die Kamera sichtbar geschaltet und anschließend gerendert (Abb. 51). Zu visualisierende Innenräume werden in zwei unterschiedlichen Auflösungen benötigt. Das Plugin muss dazu die benötigten Elemente selektieren und anschließend einen zweiten Renderingdurchlauf starten. Das Rendering von Animationen muss zurzeit manuell durchgeführt werden. Um animierte GIFs zu erstellen, müssen zuerst die benötigten Frames separat gerendert und als GIF Dateien gespeichert werden. Die Animation wird in einem Bildbearbeitungsprogramm wie zum Beispiel Photoshop nachträglich erzeugt. Zudem werden Geschosse, die in der gezoomten Ansicht eine 360 Grad Drehung erlauben sollen, auch noch einmal extra mittels einer rotierenden Kamera gerendert. Gewöhnlich sind 25-35 Frames ausreichend, um eine flüssige Rotation zu gewährleisten. Die Bilder werden durch das Plugin nach einer vordefinierten Konvention benannt und in einem ausgewählten Ordner gespeichert. Alle

statischen Grafiken werden als PNG Dateien erstellt, wobei der Hintergrund transparent gesetzt wird. Die Frames für Animation oder Rotation werden hingegen als GIF Dateien (auch mit transparentem Hintergrund) gespeichert. Dadurch das GIF Dateien nur eine Farbe als Transparent kennzeichnen können, kann es zu Qualitätsverlusten an den Rändern kommen, diese sind aber aufgrund der Rotation (=Animation) subjektiv weniger sichtbar als bei statischen Grafiken.

Nähere Beschreibung der Visualisierung

Bereiche der Präsentation wurden zuerst funktional definiert, bevor über ihre Darstellung auf der Bildschirmfläche entschieden wurde (Abb.52). Im Mittelpunkt der Präsentation (Hauptbereich) steht die Darstellung der Gebäude. Diese sollte möglichst viel Platz auf dem Bildschirm einnehmen. Die dargestellten Grafiken sollten aber auch andererseits nicht zu groß sein, um das Scrollen im Browser zu minimieren. Die Explosion läuft im vorgestellten Prototypen in vertikaler Richtung, auch dafür wurde zusätzlicher Platz über den Bildern berücksichtigt. Die meisten Interaktionsmöglichkeiten sind durch eine direkte Manipulation der Modellbilder zu betätigen. Für die andere wurde eine Navigation aufgebaut. Diese ist seitlich der Bilder platziert, um mehr Raum für die Präsentation zu haben. Der Hauptteil der Navigation ist der Platz, wo der Benutzer zwischen



Abb. 52: Interface

den verschiedenen Visualisierungsstilen wechseln kann. Über dieser Navigation befindet sich der Titel und die kurze Präsentationbeschreibung.

(Multilayer-System) Alle Teilbilder haben die gleiche Größe (gleiche Länge und Breite) und werden mit derselben Kamera aufgenommen. Ihre Position wird zuerst in einer CSS Datei definiert. Die Bilder sind dabei so organisiert, dass Bilder von unteren Geschoßen am hintersten dargestellt werden (Painter's Algorithm, Abb. 53). Dazu wird eine künstliche Tiefeninformation mit jedem Bild gespeichert, die Visualisierung übernimmt die Rolle eines Layer-Managers, der die Bilder richtig übereinander anzeigt. In der Präsentation wurden Animationen durch Animierte GIFs erstellt und als zusätzlicher Layer integriert. Einfache zu animierende 2D-Grafiken wie animierten GIFs sind im Vergleich zu Filmeinbettung die speicherschonendere Methode für die Anzeige von bewegten Daten. Der Zuseher kann eine gerade laufende Animation dabei jederzeit unterbrechen, dies geschieht durch einen Klick auf die Animationsfläche oder die

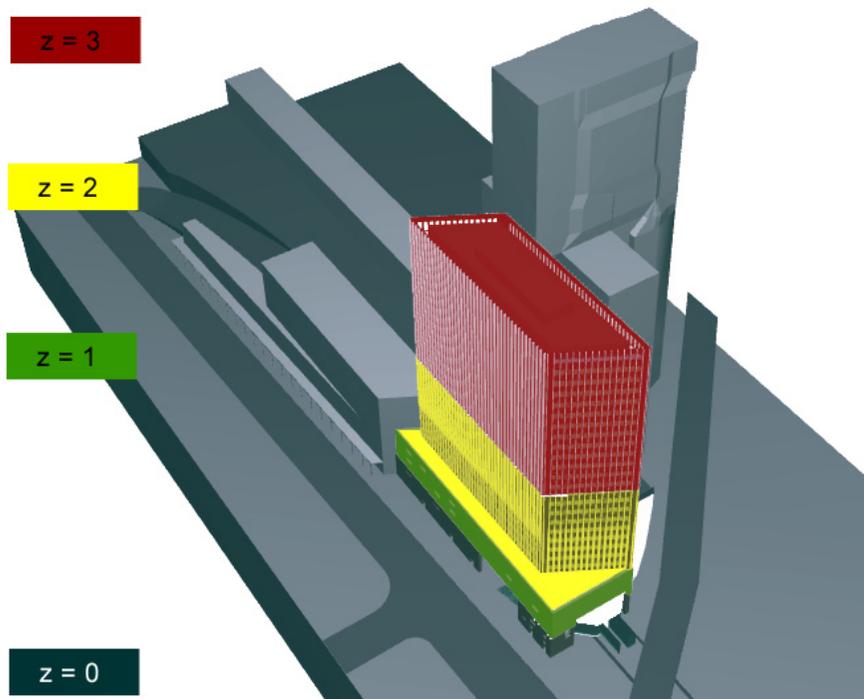


Abb. 53: Hauptlayer-System

Navigation. Technisch gesehen wird dies durch Ausblenden der animierten GIFs erreicht, weil sich diese sobald sie abgespielt werden nicht mehr unterbrechen lassen. Anstelle der animierten GIFs wird dann eine statische Grafik eingeblendet.

(Explosion Feature) Um Innenräume zu zeigen, müssen alle Teile des Gebäudes nach oben verschoben werden. Die Hintergrund (oft eine Skyline) ist dabei die einzige Grafik, die immer stationär bleibt (Abb.54). Bei Exploded Views in der Architektur sollte die Explosionbewegung wegen des einfachen Verständnisses in der Vertikalen stattfinden. Die Teile der Gebäude sollen immer in positiven Richtung der Y-Achse verschoben werden. Beim Zurückschieben sollen die Gebäudeteile in der negativen Richtung der Y-Achse bewegt werden. Der Zusammenhang der Gebäude wird durch Guides hergestellt. Die Vertikale Erschließung wird durch Liftschächte angedeutet. Das Ausmaß der Bewegung der Explosion kann linear (alle Gebäudeteile legen den selben Weg zurück, wenn sie gezogen werden) oder exponentiell (untere Gebäudeteile bewegen sich

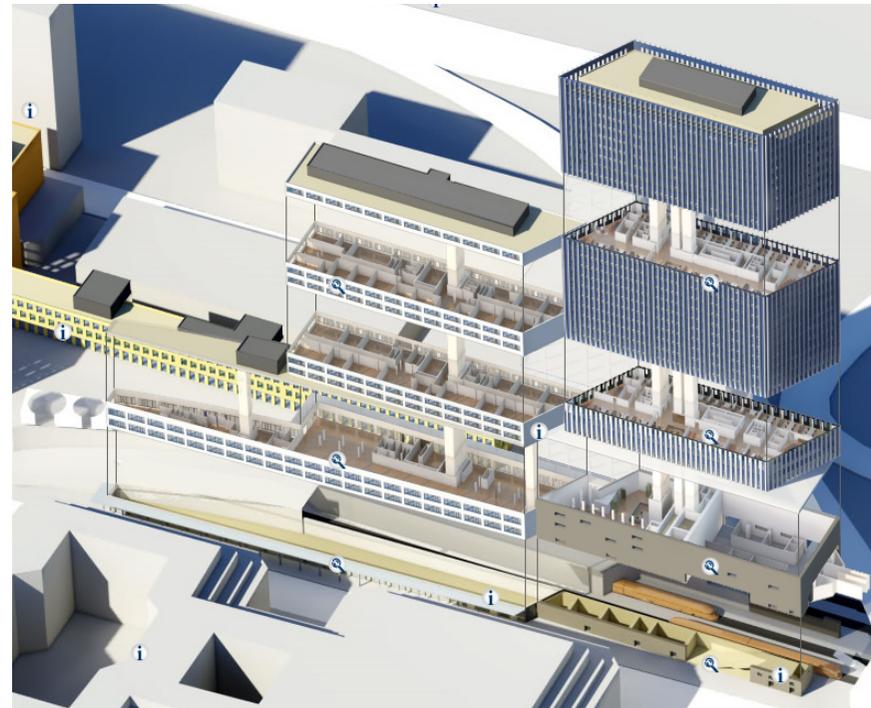


Abb. 54: Explosion Feature

weniger stark als obere Gebäudeteile) verlaufen. In der Visualisierung wurde letzteres verwendet. Die Interaktion wurde durch JavaScript erzeugt und ähnelt einem ‚Drag and Drop‘ Effekt. Die Bewegung wird auf die vertikale Richtung begrenzt. Eine Verschiebung der Gebäudeteile unter die Bodenplatte (bzw. die Hintergrundgrafik) wurde programmatisch verhindert.

(Explosion von mehreren Gebäuden) Bei einer Explosion einer Liegenschaft oder Gebäudegruppe ist es vorteilhaft dem Benutzer zu erlauben, den Innenraum von mehreren Gebäuden gleichzeitig zu sehen. Um das zu erzeugen, wurden explodierbare Bereiche für alle Gebäude definiert. Wenn sich Gebäuden überlappen, kommt das Gebäude das näher zur Kamera liegt bei der Explosion zum Zug.

(Cut-Away und Ghosted Views Features) Durch die explodierte Darstellung wurden die meisten Verdeckungsprobleme gelöst. Trotzdem kann es sein, dass einige Elemente von anderen verdeckt bleiben, dies kann durch Cut-Aways und Ghosted Views in den folgenden Form gelöst werden:

- Elemente, die durch die Außenwände verdeckt sind, können ein- und ausgeschaltet werden.
- Komplexe Grundrissabwicklungen mit unterschiedlichem Geschosshöhen und großer Zahl an Zwischengeschossen können durch einen vertikaler Schnitt sinnvoller gezeigt werden als beim Einsatz mehrerer horizontale Schnitte.
- Elemente die durch Innenraumwände verdeckt sind können durch Ghosting semitransparent dargestellt werden.
- Gebäude die durch andere Bauteile verdeckt sind können in einer Phantom View angezeigt werden. Dabei wird das vorne liegende Gebäude transparent gehalten (Abb. 55). Jedoch ergeben sich Interaktionsschwierigkeiten, wenn der Benutzer durch das transparenten Gebäude hindurch klicken will, um das hintere Gebäude zu manipulieren. Um diese zu umschiffen, wurden Griffe auf den Gebäudeteilen angebracht, die verschiebbar sein sollen. Die Griffe wurden so positioniert, dass keine visuelle Doppeldeutigkeit entsteht.

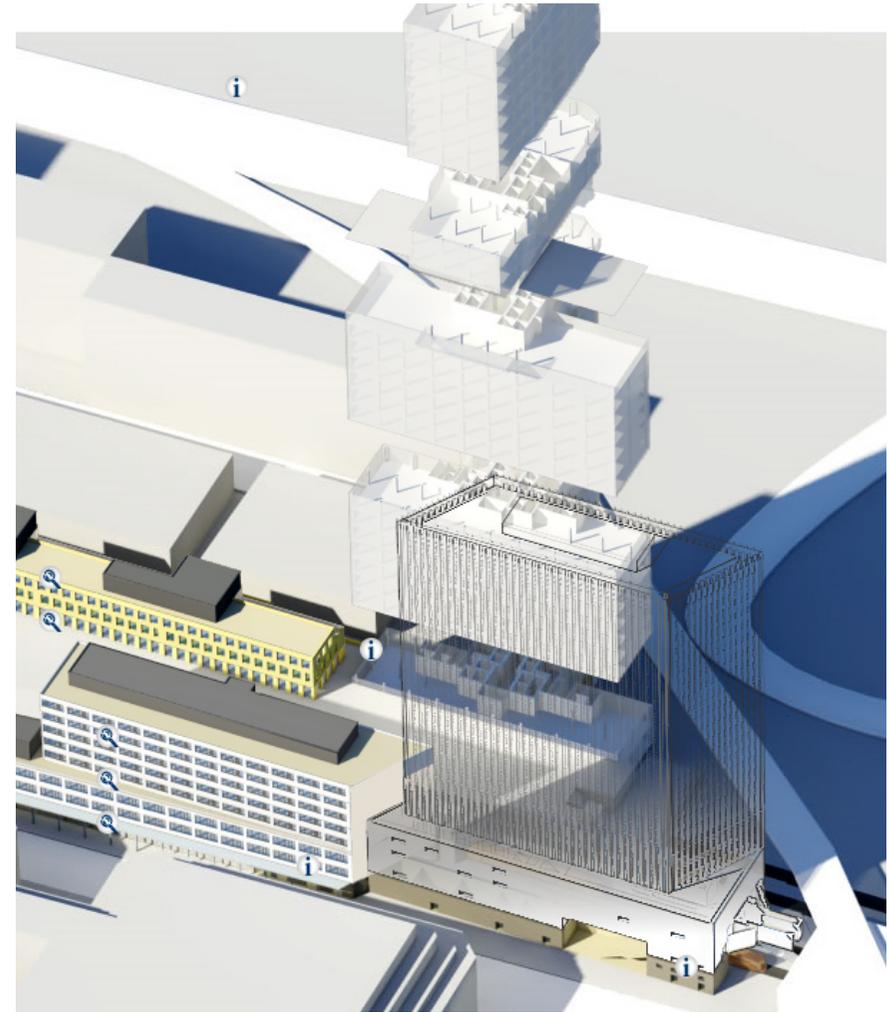


Abb. 55: Ghosting Feature (Phantom View)

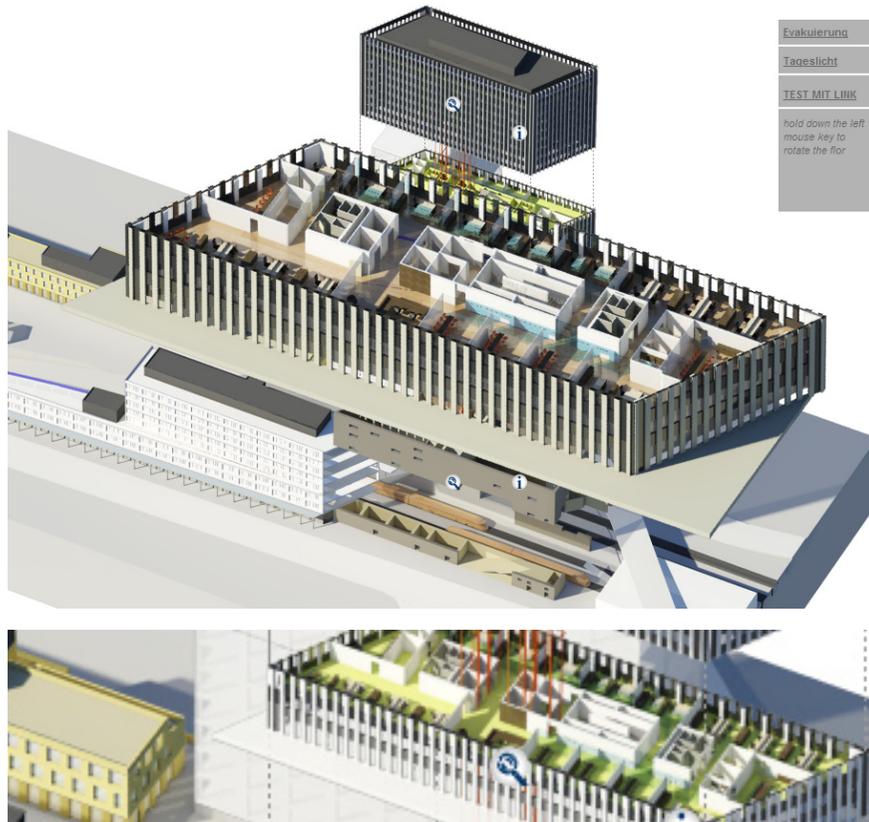


Abb. 56: Zoom Feature

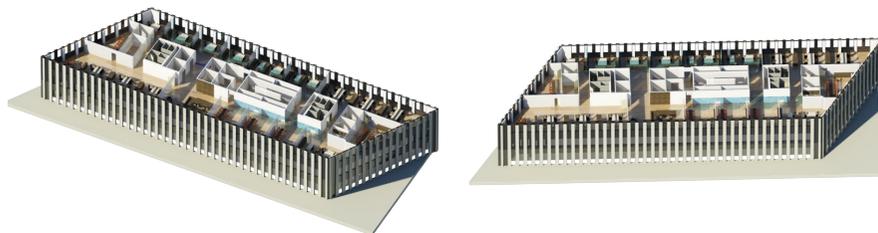


Abb. 57: Frames für den Rotation Feature

Von der technischen Umsetzung her sind Cut-Aways und Ghosted Views im Prototypen einzeln integriert. Bei Cutaways wurde durch die Navigation die Möglichkeit angeboten, einen Teil eines Objekts aus- und einzuschalten. Beim Rendering in 3DS Max wurden diese Teile der Gebäude als getrennte Grafiken gespeichert und durch JavaScript selektiv gezeigt. Für Ghosted und Phantom Views wurden die Grafiken in Photoshop nachgearbeitet um Transparenzen zu erzeugen. Zur Laufzeit werden diese Grafiken dann wechselweise eingeblendet.

(Zoom Feature mit optionaler Rotation) Um eine Explosion an einem mehrgeschossigen Gebäude weitgehend ohne Scrollen zu ermöglichen und zudem die relevante Umgebung darzustellen muss die Szene skaliert werden. Dabei kann es bei komplexeren Szenen sein, dass relevante Innenräume schlecht sichtbar bleiben. Um dieses Problem zu beheben, ist es durchaus sinnvoll, eine Zoom Funktion an gewissen Gebäudeteilen zu ermöglichen. Es gibt unterschiedliche Möglichkeiten, um diese Zoomfunktion zu erstellen: Die Detailansicht kann direkt über dem gezoomten Gebäudeteil angezeigt werden, oder in einem eigenen Zusatzbereich neben der Visualisierung. Für den Prototypen wurde erstere Möglichkeit genommen. Als Symbole für das Zoomfeature wurden kleine Icons (Vergrößerungsgläser) auf die Gebäudeteile platziert. Diese Grafiken können geklickt werden und dadurch wird die Detailansicht angezeigt (Abb.56). Um Wartezeiten bei dem Laden zu vermeiden, wurden alle diese zusätzlichen Detailgrafiken im Vorhinein geladen und versteckt. Erst beim Klick wird durch ein Skript die richtige Grafik ausgesucht, richtig positioniert und zur Anzeige gebracht. Im Prototypen wird in der Detailsicht eines Geschosses die Möglichkeit der Rotation geboten (Abb. 57). Die Rotation ist durch ein eigenes Plugin für JQuery erstellt worden, dass einen fließenden Übergang zwischen zwei Grafiken ermöglicht.

(Labels Vordefinieren) Bei explodierten Darstellungen ist es oft notwendig, bestimmte Modellteile zu beschriften und zusätzliche Informationen wie Fotografien oder Grafiken anzuzeigen. Dies wurde durch ein eigenes JavaScript Labeling System umgesetzt (Abb. 58). Die Informationen für die Labels können in beliebige HTML-Elemente geschrieben und an beliebige Positionen der Visualisierung platziert werden. Das Layout aller Labels wird vom JavaScript Code und einer zusätzlichen CSS Datei gesteuert.



Abb. 58: Labels

Lösung der Probleme Dreidimensionaler Abbildungen in der vorgestellten Arbeit

Die eingangs erwähnten Probleme der Verdeckung, Zuordenbarkeit der Innen- und Außenräume, Sichtbarkeit des Konstruktionssystems sowie sonstiger Metainformationen (P1-P4 auf Seite 15) werden in der vorgestellte Visualisierung wie folgt behandelt:

- L1: Verdeckungen werden durch Explosion beziehungsweise Möglichkeit des Ein-/Ausblendens, Cut-Aways und Ghosted Views weitgehend gelöst. Dabei kommt es auf die Intention des Architekten an, welchen der Techniken er für ein gesetztes Ziel benutzt. Wir haben uns entschieden, zusätzlich zu Explosionsmöglichkeiten für alle Gebäude, die weiter hinten liegenden Bauteile solide und die vorne liegenden Bauteile als Phantom Views abzubilden. Weiters dient die Interaktivität dazu, Gebäudeteile zu verschieben und sich so ein freies Blickfeld zu verschaffen.
- L2: Innenraum und Außenraum können klar zugeordnet werden: Durch die Explosionsdarstellung ergibt sich eine Innen-/Außenbeziehung zwischen den Gebäuden und ihren Geschossen, welchen dem Benutzer erlaubt diese mental auseinander zu halten.
- L3: Das Konstruktionssystem kann auf Wunsch eingeblendet werden. Obwohl in diesen Projekt eher das bauphysikalische System von Interesse war, könnte jederzeit auch ein eigenen Layer für das Konstruktionssystem eingliedert werden
- L4: Metainformationen wie das Raumkonzept können und wurden als Layers inkludiert. Zusatzinformationen wie zum Beispiel Beschriftungen und symbolische Darstellungen von Gebäudetechnik wurden als Labels hinzugefügt.

5. Schlussbetrachtung

In dieser Diplomarbeit wurden expressiven Darstellungen als Möglichkeit, Innen- und Außenräume gemeinsam darzustellen, präsentiert. Dabei sind die Grundsätze der Exploded Views, Cut-Aways, der Ghosted Views und weiterer expressiver Ansätze erklärt worden. Das ausgearbeitete Beispiel zeigt einen interaktiven Einsatz von expressiven Darstellungen in einer Architekturpräsentation. Als Zusatznutzen zur herkömmlichen Architekturvisualisierung wurde ein Gewinn an Informationen durch die bewusste räumliche Änderung der Gebäudeteile, Abstraktionen und Interaktivitäten identifiziert. Das Erkunden der Architekturräume in einer interaktiven Visualisierung bietet den Vorteil, räumliche Zusammenhänge noch besser begreifen zu können. Innere Gebäudesysteme, wie das bauphysikalische System, die Statik, das Raumprogramm oder die Erschließung werden durch ein-/ausschaltbare Layer besser vermittelt als durch getrennt vorliegende Grafiken. Zudem wird die Beziehungen zwischen den Modellteilen durch Guides gezeigt. Die erwähnten Probleme der Verdeckung, Zuordenbarkeit der Innen- und Außenräume, Sichtbarkeit des Konstruktionssystems sowie sonstiger Metainformationen wurden durch die DA behandelt und durch expressiven Techniken weitgehend gelöst.

Mögliche Einsatzgebiete

Eine **Darstellung der Wegführung** könnte bei Schulen, Universitäten oder Öffentlichen Gebäuden eine leichtere Orientierung vor dem ersten Eintreten komplexer Gebäude bieten. Für ein öffentliches Gebäude würde diese Art der Internet Präsentation ein praktisches Verständnis der räumlichen Verteilung der Abteilungen ermöglichen. Die Internetpräsentation könnte als Knoten der Bauanlage verwendet werden, in der alle Abteilungswebseiten verlinkt würden. Damit würde dem Benutzer ein räumlicher Gebäudeplan angeboten mit zusätzlich wichtigen Informationen (beispielsweise Kontaktdaten, Öffnungszeiten, Abteilungen) angeboten werden.

Komplexe Gebäudekonstruktionen können durch die beschriebene Darstellungsform von einem Benutzer auseinandergenommen und dann wieder zusammengestellt werden. Dieser spielerische Ansatz, könnte helfen, leichter räumliche Beziehungen zu verstehen. Damit könnten Architekten oder Bauingenieure leichter komplexe Systeme kommunizieren und vorstellen.

Interaktive expressive Darstellung könnten für **Immobilienvorstellungen** vor potenziellen Kunden verwendet werden. Solche Präsentationen über das Internet könnten leicht eine große Zahl an Interessenten erreichen. Die Benutzer würden dadurch eine klarere Vorstellung der Räumlichkeiten, der Umgebung und der Zusammengehörigkeit bekommen und sich in der Folge vielleicht auch leichter für eine Immobilie entscheiden. Als Zusatzinformation könnte dabei die Dimensionen der Räume und ein Maßwerkzeug angeboten werden.

Diese Art von Interaktiver Visualisierung könnte in beliebigen weiteren Bereichen verwendet werden, um mehr Informationen von einem Gebäude oder einer Bauanlage darzustellen (**Informationsvisualisierungen**). Museen könnten zum Beispiel eine erste Vorstellung der Räumlichkeiten mit Zusatzinformation für ausgestellte Werke online anbieten. Im Bereich der Architektur könnten solche Präsentation weiter für Darstellungen von Ideen (Konzeptionsphase) verwendet werden.

Möglichkeiten für Weiterentwicklung

Diese Interaktiven Expressiven Visualisierungen für die Architektur könnten mehrfach erweitert werden. Bereiche für eine weitere Untersuchung wären:

- **Rotation bestimmter Gebäudeteile:** Gebäude können mit derzeitigem Stand grundsätzlich nicht rotiert werden. Eine Kamera wird vorab eingestellt und für die gesamte Präsentation verwendet. Durch die Navigation kann derzeit eine Möglichkeit angeboten werden, die ganze Präsentation in der Nord-, West-, Ost- und Südseite zu zeigen. Mit einer rotationellen Erweiterung könnten verdeckte Gebäudefassaden in den Vordergrund gebracht werden, dafür ist aber eine Wiederholung des ganzen Erstellungsprozesses für jeden anzuschauenden Winkel notwendig. Um das zu ermöglichen müsste das Plugin erweitert werden, um ein vollautomatisches Rendering aller benötigten Grafiken zu ermöglichen.
- **Zielexplosion:** Bei explodierten Darstellungen wäre es praktisch, nur die Teile der Gebäude zu verschieben zu können, die von Interesse sind. Eine Teilexplosion könnte hier eine Alternative zu einer voll explodierten Darstellung sein. Der Benutzer könnte mit Mausclick das Gebäudeteil aussuchen, zu dem der Innenraum gezeigt werden soll und dieses dann einzeln verschieben.
- **Maßwerkzeug:** Die Dimensionen der Gebäude können in der Präsentation derweil nicht gemessen werden. Es kann zwar die Nutzfläche in den Bildern oder mittels Labels ausgegeben werden, ein eigener Bemaßungslayer scheint aber weitaus sinnvoller.
- **Gebäudesuche:** Eine der meistverwendeten Funktionen des Internets ist das Durchsuchen von Dokumenten, Seiten oder Bilder. Eine ähnliche Funktion in der Visualisierung könnte angeboten werden, womit die Gebäude beispielsweise nach bestimmten Räumlichkeiten durchsuchbar gemacht werden könnte. Dafür können alle Teilbilder durch Beschreibungen oder Schlüsselwörter versehen werden. Diese Texte können dann durchgesucht werden und das Ergebnis hervorgehoben bzw. mit einer zusätzlichen Grafik markiert werden.

Eine andere Möglichkeit wäre es, die Beschreibungen der Labels durchzusuchen und die zugehörige Grafik zu markieren. Das könnte zum Beispiel benutzt werden um eine Suche nach einem bestimmten Hörsaal in einem dargestellten Universitätsgebäude zu ermöglichen. Diese Durchsuchmöglichkeit könnte noch einfacher durch einen Zusatzlayer mit Grafiken und eine Liste von Bereichen aus- und eingeblendet werden. Für eine Wohnimmobilie könnte beispielsweise eine Liste mit allen unterschiedlichen Wohnungsgrößen eingefügt werden. Bei einer Auswahl von einer gewünschten Größe könnte durch Teilexplosion nur diese bestimmte Räumlichkeiten gezeigt werden. Die gesuchten Apartments sollten bestenfalls noch zusätzlich angedeutet werden.

Anhang A: Architekturvisualisierung für das Web

Das Internet ermöglicht einen bequemen allgemeinen Datenaustausch zwischen Firmen, staatlichen Organisationen und privaten Anbietern. Wesentlicher Vorteil gegenüber anderen Medien ist die kurze Dauer des Informationstransports und die Einsparung von Druck- und Verteilungskosten. Ein großer Vorteil des Internets ist der **multimediale Aspekt** und die vielfältige Möglichkeit, Informationen darzustellen. Komplexe Abläufe oder Objekte, die lange und genau mit Hilfe von Text und Bildern beschrieben werden müssen, können durch Grafiken oder Animationen sehr anschaulich dargestellt werden. Statische Webseiten können mit 2D Grafiken, 3D- Modellen, Animationen und Videos ergänzt werden, um Projekte vorzustellen. Die Grafikformate für das Web lassen sich genau wie alle anderen Grafiken in zwei-, drei- sowie vierdimensionale Grafiken unterteilen.

2D: Vektor- und Rastergrafiken

Im Vergleich zu den Grafiken für Printmedien werden die Bilder für das Internet auf Bildschirmdarstellung (72dpi) optimiert. Dabei sollen Ladezeiten möglichst gering gehalten werden. Kriterien für die Auswahl von Grafikformaten für das Web sind die Browserunterstützung, das Kompressionsverfahren, das Farbmodell und die Animationsmöglichkeit. Grafiken lassen sich weiter in Raster- und Vektorgrafiken unterteilen.

Eine Vektorgrafik ist aus Linien, Kurven und Flächen aufgebaut, die mathematisch genau beschrieben werden können. Diese können ohne Qualitätsverlust stufenlos vergrößert, verkleinert, gestaucht und verzerrt werden.

Die Eigenschaften einzelner Linien, Kurven oder Flächen bleiben erhalten und lassen sich nachträglich ändern. Die meistverwendete Vektorgrafiken zurzeit im Internet sind das offene Format SVG (Scalable Vector Graphics) und die SWF-Dateien von Adobe Flash. Diese können mit Hilfe von Plug-Ins angeschaut werden.

Eine Rastergrafik oder Pixelgrafik ist im Gegensatz zur Vektorgrafik ein Bildformat, das aus einzelnen Pixeln besteht. Dies ist auch ein Grund für die schlechten Kompressionseigenschaften dieses Formates. Rastergrafiken eignen sich für komplexe Bildstrukturen wie Fotos, bei geometrischen Figuren oder Schriften kommen bevorzugt Vektorgrafiken zum Einsatz. Rastergrafiken für das Internet werden in der RGB-Farbraum gespeichert. Im Gegensatz zu Vektorgrafiken können Rasterformate nicht unter Beibehaltung der Qualität vergrößert werden. Im Web haben sich das JPG und den GIF Format durchgesetzt. Das derzeit noch wenig verbreitete PNG Format vereint die Stärken der beiden anderen Grafikformate. Es werden nun die Charakteristiken und Einsatzgebiete dieser Formate gegenübergestellt und in Tabelle 1 zusammengefasst.

	GIF	JPEG	PNG
Transparenz	1 Bit	-	1 / 8 Bit
Animation	Ja	-	-
Farbmodus	indexiert	RGB	indexiert/RGB
Farbtiefe	bis 8 Bit	24 Bit	bis 24 Bit
Kompression	verlustfrei	verlustbehaftet	verlustfrei
Interlacing	Ja	Progressive JPEG	Ja

Table 1: Gegenüberstellung von Grafikformaten

(Graphics Interchange Format) Das GIF Format ist verlustfrei komprimierbar. Dadurch wird die Dateigröße verkleinert, ohne dass das Bild an Auflösung oder Schärfe verliert. GIF ist aber in der Farbtiefe auf 256 Farben limitiert, was einem 8 Bit indizierten Farbbild entspricht. Die Kompression bringt insbesondere Vorteile in Dateigröße und Qualität, wenn das Bild wenige Farben verwendet (wie z.B. Strichgrafiken, Logos oder Illustrationen mit Text). Hierbei wird von 1 bis 8bit komprimiert. GIF ist für Fotos und Farbverläufe wegen der begrenzten Farbtiefe weniger geeignet als z.B. JPEG. Im optischen Vergleich sieht man den Unterschied zwischen einem gespeicherten Foto im JPG und GIF-Format (Abb. 59). Das GIF Format unterstützt Interlacing, was sehr vorteilhaft für das Web ist: Bei der Anzeige der Grafik im Webbrowser werden die Zeilen in zunächst 8-facher, dann 4-facher, schließlich doppelter Höhe angezeigt, bis alle Daten geladen sind und das Bild 1:1 dargestellt wird¹⁵. Im Web bringt das GIF die Möglichkeit der Transparenz mit sich. Ein Farbeintrag kann in der Palette als



Abb. 59: a) JPEG Grafikformat; b) GIF Grafikformat

¹⁵ aus Wikipedia „GIF“, http://de.wikipedia.org/wiki/Graphics_Interchange_Format#Interlacing, 18. Oktober 2009

transparent definiert werden, dadurch kann die jeweilige Hintergrundfarbe an gewünschten Stellen durchscheinen. Die Transparenz ist nur binär, wobei ein Pixel entweder nur „voll sichtbar“ oder „komplett durchsichtig“ sein kann. GIFs können auch animiert werden, indem mehrere einzelne GIFs zusammengesetzt in ein einziges GIF gespeichert werden. Dabei könnte die Abspielgeschwindigkeit und die Wiederholungszahl unterschiedlich angegeben werden.

(Joint Photographic Expert Group) JPG oder JPEG stellt bis 16.7 Mio Farben dar, was der Farbtiefe eines 24bit Bildes gleicht und in diesem Punkt besser als GIF ist. Dieses Format ist das richtige für die Komprimierung von Halbtonbildern, wie z.B. Fotos. Die Größe/Qualität könnte zwischen 0% und 100% verlustbehaftet komprimiert werden. Zwischen 80% und 100% ist ein Bild vom Original wenig zu unterscheiden, wogegen die Dateigröße bereits massiv kleiner ist. Ähnlich wie beim vom GIF-Format bekannten Interlacing gibt es hier auch eine Möglichkeit, zuerst eine undeutliche Version des Bildes am Bildschirm anzubieten und diese dann nach und nach zu vervollständigen (progressive JPEG). Der Nachteil von JPGs ist, dass diese weder transparent sein können noch animierbar sind.

(Portable Network Graphic) Dieses Grafikformat lässt sich als 8 oder 24bit (PNG-8 oder PNG-24) verlustfrei komprimiert abspeichern. PNG-24 sind im Vergleich zum JPEG-Format wesentlich größer, weil PNG-24 ein verlustfrei komprimierbares Format darstellt. Zusätzlich verfügt PNG über eine Möglichkeit zur progressiven Darstellung und dem Interlacing. Dabei treten weniger Bildverzerrungen auf als beim GIF-Format und das erste Vorschau-Bild wird schon angezeigt, wenn nur wenige Prozent der Gesamtdatei geladen wurden. Das PNG-Format kann neben den reinen Bildwerten noch zusätzliche Informationen speichern, wie z.B. Bildherkunft, oder Gammawert. Ein isoliertes Laden von Bildteilen ist hingegen nicht möglich. PNG-Dateien unterstützen 1Bit (binärer an/aus) oder 8 Bit Transparenz Modus (Alphakanal). Das PNG-Format kann durch diesen Alphakanal unabhängig vom Hintergrund die Kanten von Text und Bildern glätten. Die Darstellung von PNG-Dateien mit Alphakanal (8 Bit Transparenz) ist in manchen älteren Browsern (namentlich IE 6 und davor) nicht korrekt implementiert. PNG-Dateien mit binärer Transparenz hingegen werden fehlerfrei dargestellt. Zur Behebung dieser fehlerhaften Darstellung des Alphakanals in

älteren Versionen des Internet Explorers gibt es aber auch Umgehungs-lösungen. Von einigen Entwicklern sind Java Script Lösungen geschrieben worden, womit die Transparenzfähigkeit der PNG in älteren Versionen des IE ermöglicht wird. PNG bietet keine Möglichkeit zur Animation wie das GIF-Format.

In der Webentwicklung tritt immer wieder die Frage auf ob die PNGs verwendet werden sollen. Dies hat damit zu tun, dass früheren Browser dieses Format nicht unterstützt haben. Aus der angeführten Statistiken (Abb. 60, 61) kann man sehen, dass die Benutzer der älteren Browsergeneration des Internet Explorer in der ersten Hälfte des Jahres 2009 weltweit um 15% und in Österreich um 10% gefallen sind. Es ist auch zu bemerken, dass der Wert jedes Monats weniger wird. Daher kann mit Sicherheit gesagt werden, dass nur noch wenige Leute auf solche alte Browser setzen und der Einsatz des PNG ohne weiteres möglich ist.

2009	IE7	IE6	IE8	Firefox	Chrome	Safari	Opera
June	18.7%	14.9%	7.1%	47.3%	6.0%	3.1%	2.1%
May	21.3%	14.5%	5.2%	47.7%	5.5%	3.0%	2.2%
April	23.2%	15.4%	3.5%	47.1%	4.9%	3.0%	2.2%
March	24.9%	17.0%	1.4%	46.5%	4.2%	3.1%	2.3%
February	25.4%	17.4%	0.8%	46.4%	4.0%	3.0%	2.2%
January	25.7%	18.5%	0.6%	45.5%	3.9%	3.0%	2.3%

Abb. 60: Browserstatistiken - Weltweit

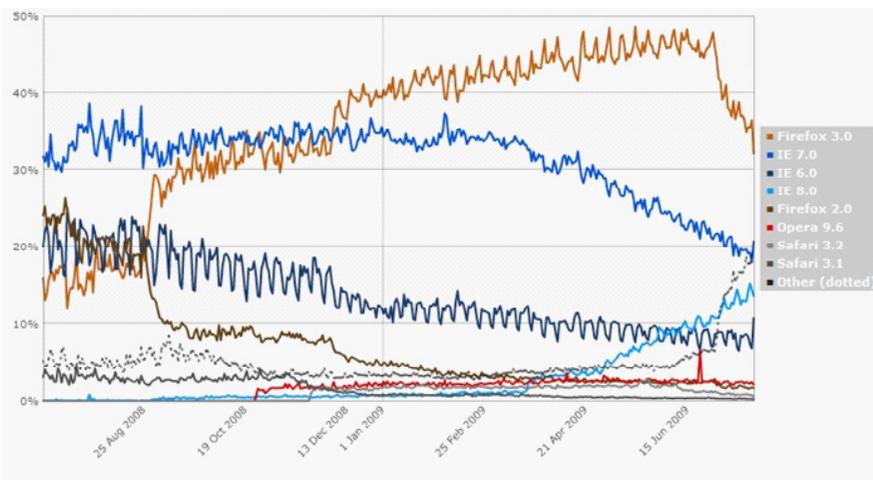


Abb. 61: Browserstatistiken - Österreich

3D: Geometrisches Modell

Im Web ist die Verwendung von dreidimensionalen Modellen für Architekturvisualisierungen sehr selten. Der Vorteil solcher 3D Grafiken liegt in der Freiheit, das Modell virtuell im Web von jedem beliebigen Standpunkt aus zu betrachten und es bietet damit einen höheren Level an Interaktivität als andere Grafiken. Dreidimensionale virtuelle Objekte werden in einem beliebigen 3D Modellierungsprogramm (wie z.B. 3D Studio Max, Blender, Google SketchUp oder Cinema 4D) erstellt und danach in ein internettaugliches Format exportiert. Durch Autorenprogramme könnten zusätzlich Interaktionsmöglichkeiten bestimmt und kontrolliert werden. Eine Standard Beschreibungssprache für 3D-Szenen für das Internet ist das X3D (Extensible 3D) als Erweiterung des vormals standardisierten VRML-Format, sowie einigen Formaten die mit dem Einsatz des Webs für den Austausch von 3D-Daten verbreitet wurden (Collada, KML).

Animation und Video

Architekturvideos im Web sind eine Darstellungstechnik für Dokumentation geplanter und gebauter Architektur. **Animation** wird verwendet, um das zeitabhängige Verhalten von komplexen Systemen zu zeigen (z.B. das „Durchwandern“ oder „Durchfliegen“ eines Gebäudes). Dabei gibt es größere Möglichkeiten zur Steuerung der Einzelbilder und ihren Durchlauf. Die meistverwendeten Techniken sind die Animierten GIFs, Dynamisches HTML, sowie Flash. Aber auch dreidimensionale Animationstechniken sind am Web im Einsatz, hier wird kurz das X3D Format vorgestellt.

(Extensible 3D) Extensible 3D ist ein offener Standard für virtuelle 3D-Modelle, der auf XML Technologie aufbaut. Der bildet den offiziellen Nachfolger des VRML-Standards und ist seit 2004 als ISO-Standard spezifiziert. Genau wie in VRML lassen sich auch in X3D dreidimensionale Zeichnungen, wissenschaftliche Visualisierungen, interaktive Lernanwendungen oder Spiele in Echtzeit realisieren. X3D unterstützt 2D- und 3D-Graphik, Animation, Audio, Video, CAD-Daten, Scripting, Physikalische Simulationen und Echtzeitkommunikation. Gegenüber VRML stehen jedoch bei X3D wesentlich mehr standardisierte Möglichkeiten und Schnittstellen bereit. Das Hauptaugenmerk bei XML liegt in der Trennung von Inhalt und Form. Für die Darstellung der 3D-Szene von

einer X3D-Datei wird ein Browser Plugin benötigt, wie zum Beispiel den BS Contact VRML/X3D, Open VRML, Octaga FreePlayer, FreeWrl, oder Webbrowser-unabhängige Programme wie Xj3D.

Durch X3D kann die Verhaltensweise von Objekten definiert werden. Weiters bietet sich die Möglichkeit, eine Verbindung zu externen Applikationen mittels Programmier- und Skriptsprachen zu erstellen. Der Aufbau von X3D ist streng modular (24 Komponenten und weitere Unterteilung durch Ebenen). Aus den Komponenten werden Profile zusammengestellt. Zum Beispiel ist das derzeit in Entwicklung befindliche CDF-Profil (CAD Distillation Format) für die Übersetzung von CAD-Daten in ein offenes Format gedacht.

A.1. Technologien zur Erzeugung von interaktiven Visualisierungen im Web

Interaktiv bedeutet „wechselseitig“ und „aufeinander bezogen“. In der Visualisierung ist eine Wechselbeziehung zwischen einer Computer Applikation und einem Benutzer gemeint. Die Interaktion findet dabei über alle vorhandenen Kanäle statt, wie Bildschirm, Maus, Tastatur, Lautsprecher usw. Sie soll vom Ersteller vorher genau geplant werden, damit die Informationen bestmöglich wahrgenommen und interpretiert werden zu können. Bei solcher Kommunikation zwischen Benutzer und Applikation kann die Dialogkontrolle beim Benutzer, beim System oder hybrid sein. Das System könnte Eingaben anfordern und nachher verarbeiten, wobei es keinen direkten Einfluss vom Benutzer gibt. Das ist eine systemgesteuerte Interaktion. Der Benutzer könnte aber bei einer Benutzergesteuerter Interaktion Aktionen initiieren, die nach ihrer Durchführung Ergebnisse zurückmelden. Bei Mischformen wechselt die Kontrolle zwischen

dem System und dem Benutzer. Der Interaktionsablauf sollte möglichst benutzergesteuert sein, um dem Mensch Freiheit in der Informationsansicht zu geben¹⁶.

Die Interaktion zwischen Benutzer und einer Webseite ist generell sehr einfach und erfolgt durch klicken von Links. Mit der Entstehung neuer Internettechnologien ist es zunehmend möglich, die bei Desktop-Programmen übliche Interaktion auch bei Online-Anwendungen zu erwarten. Im Vergleich zu statischen Webseiten muss sich der Benutzer dann nicht mehr durch eine Unzahl an Seiten klicken, sondern verweilt auf einer Seite, welche aussieht wie eine Desktop-Applikation. Damit entsteht ein fließendes Informationserlebnis¹⁷.

Web Applikationen residieren im Webbrowser. Alle Daten und Informationen könnten entweder auf einmal geladen werden oder sie werden schrittweise nachgeladen. Bei ersterer Variante entsteht eine gewisse Wartezeit zu Beginn, das Benutzungserlebnis ist aber in der Folge flüssig und interaktiv. Im Zweiteren Fall startet die Applikation schneller, es kann jedoch beim nachträglichen Laden von Daten zu Wartezeiten kommen, was den interaktiven Eindruck vermindern kann¹⁸.

Bei einer Architektur Webpräsentation ist zwischen **Kontrollmöglichkeiten in einen Interface** (Abb. 62a) und **direkte Manipulation der Objekten** (Abb. 62b), z.B. Rotation oder Ein/Auszoomen, zu unterscheiden. Die direkte Manipulation des Modells ist generell bei Echtzeit 3D Visualisierungen oder 2D Bildern wie zum Beispiel Panoramas zu sehen. Die zugrundeliegenden Technologien für Erzeugung von Interaktiven Darstellungen im Web sind meistens Adobe Flash, Adobe Flex, Dynamic HTML, AJAX. X3D würde bereits in den vorangehenden Kapiteln vorgestellt, es dient zum Speichern und Wiedergeben von 3D Modellen.

¹⁶ vgl. Herczeg, 2005

¹⁷ vgl. Kalbach, 2008

¹⁸ vgl. Spool; Scanlon, 1999

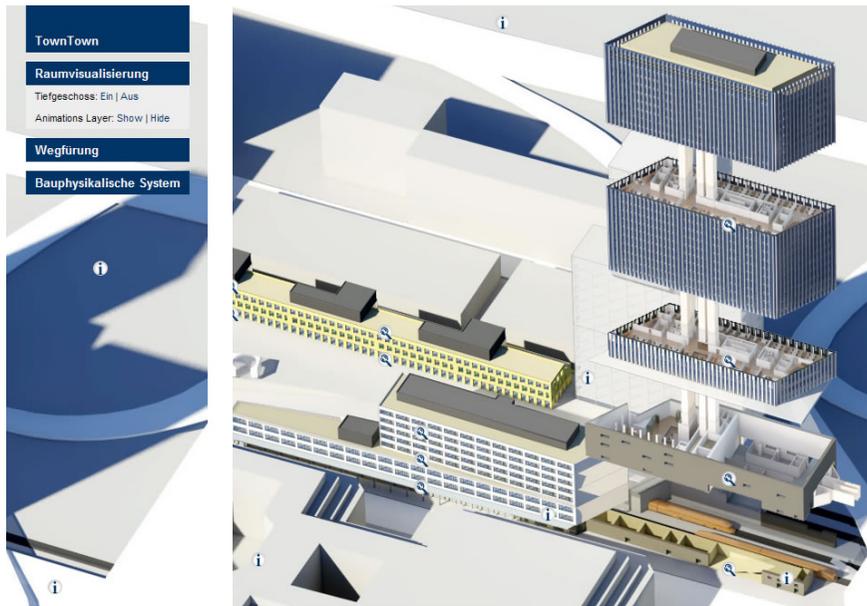


Abb.62: a) Interface; b) Direkte Modelmanipulation

(Adobe Flash und Flex) Ein bekanntes Tool zur Erstellung von interaktiven Animationen ist Flash von der Firma Adobe. Um komplexere webbasierte Anwendungen mit optisch aufwändigen Benutzeroberflächen und Animationen zu erstellen, kann auch die neuere Webtechnologie Adobe Flex eingesetzt werden. Flash wurde primär für Designer als Zielgruppe gerichtet. Dagegen bietet das Flex Werkzeug einen intuitivere Programmierung für Entwickler an. Der Vorteil beider Technologien ist die gute Komprimierungsfähigkeit. Ursprünglich war das Flash Tool als ein reines Animationswerkzeug für das Web gedacht und wurde meistens für so genannte Intros auf Webseiten oder animierte Werbeflächen eingesetzt. Durch die Erweiterung mit der Programmiersprache ActionScript ist eine Erstellung komplexer Webinhalte oder eine Programmierung browserbasierter Anwendungen möglich, zum Beispiel in Form von Kugelpanoramen, 3d Gebäudemodelle oder multimedialen Präsentationen.

Um Interaktive Applikationen erstellen zu können, ist grundsätzlich das Programm Adobe Flash oder Flex notwendig. Eine andere Vorgehensweise wäre, die Programmierung in einer XML Sprache durchzuführen und den Code

anschließend durch einen Compiler in eine ausführbare .swf Datei umzuwandeln. Um die die Anwendung in einem Browser betrachten zu können, ist ein Plugin erforderlich. Die Technologie ist nicht als offener Standard spezifiziert und es wird kritisiert, dass dies den freien Informationsaustausch im Internet behindert. Im Vergleich zu anderen ähnlichen Technologien hat der Adobe Flash Player trotzdem den größten Verbreitungsgrad. Ein anderer Vorteil der Pluginverwendung ist die Browser- und Plattformunabhängigkeit.

(AJAX und Dynamic HTML) AJAX Applikationen erweitern das traditionelle Webapplikationenmodell auf folgende Weise: Die clientseitige Arbeit übernimmt eine in JavaScript geschriebene Komponente (die sogenannte Ajax-Engine). Wenn Daten vom Server benötigt werden, zum Beispiel bei Nachladen einzelner Bausteine der Benutzeroberfläche, wird vom Server ein XML-Dokument angefordert und der Bildschirminhalt neu aufgebaut. Neben der Browser Add-On Programmiersprache JavaScript sind die Technologien HTML, CSS und DOM relevant. CSS und HTML sind zum Verfassen von Dokumenten da, JavaScript fungiert als clientseitige Skriptsprache. Die Benutzung von Javascript zur Änderung und Erweiterung von HTML Dokumenten (DOM) ist auch als Dynamic HTML (DHTML) bekannt. Der Vorteil der AJAX Technologie ist ihre Standardisierung, wodurch für das Ausführen solcher Anwendungen nur ein gängiger Browser mit aktiviertem JavaScript notwendig ist. Neue Interaktionsstile, wie zum Beispiel Drag and Drop oder Schieberiegel, werden dabei voll unterstützt. Die oft verwendete Zurück-Schaltfläche eines Browsers oder das Hinzufügen von Lesezeichen für einen bestimmten Zustand der Applikation kann jedoch nicht gewährleistet werden.

A.2. Kriterien für die Bedienbarkeit einer Webapplikation

Usability bezeichnet die Nutzungsqualität bei der Interaktion mit einem System. Die Kriterien für Webapplikationen weichen dabei von den Kriterien für eine Webseite ab und hängen auch vom Einsatzgebiet und der Zielgruppe ab. Unabhängig von der speziellen Zielsetzung und den unterschiedlichen Nutzergruppen ist bei informationsorientierten Webpräsentationen vorauszusetzen, dass die Informationen das wichtigste Anliegen der Benutzer darstellen. Aufgabe einer Webvisualisierung wäre es daher, bei der Auffindung derselben Effizienz und Effektivität zu ermöglichen. Je effizienter die jeweilige Anwendung gestaltet ist, desto schneller wird die Zielvorgabe der Benutzer erfüllt. Bei der Entwicklung einer Webpräsentation sollte auch nicht vergessen werden, dass der Entwickler selbst keinen typischen Benutzer darstellt und was für manchen bedienbar ist, muss nicht zwangsläufig auch für alle benutzbar sein. Kriterien für eine gut bedienbare Webapplikation so wie wir sie geschrieben haben sind:

- Der im Webbrowser sichtbare Inhalt soll sinnvoll gruppiert sein, so dass eine intuitive und rasche Bedienung möglich ist.
- Der Aufwand des Einarbeitens in die noch unbekannte Anwendung soll reduziert werden.
- Plug-Ins und Browsererweiterungen sollen sparsam verwendet werden. Viele Benutzer wollen keine zusätzlichen Anwendungen herunterladen müssen, um zu den gewünschten Informationen zu kommen. Das Vertrauen zu den Anbietern der Webanwendung und den Plug-In Ersteller spielt auch eine Rolle.
- Die Wartezeiten beim Laden einer Webanwendung sollten möglichst minimiert werden. Manche der Maßnahmen, um das zu erreichen, sind: der Verzicht von unnötigen Grafiken und Multimedia-Effekten, optimale Komprimierung notwendiger Grafiken oder Verwendung von Stylesheets(CSS). Wenn eine Verzögerung trotzdem zu sehen ist, sollte ein visuelles Feedback mit Ladestatus gezeigt werden.

- Die Wahl des richtigen Datenformats für die jeweiligen Medientypen ist von Bedeutung, um lange Übertragungszeiten zu vermeiden.
- Fehlermeldungen sollten verständlich sein und Hinweise über mögliche Problemlösungsstrategien enthalten. Der Nutzer muss weiters über irreversible Handlungen informiert werden.
- In Hinsicht auf eine effiziente Bedienbarkeit stellt sich die Frage nach einer idealen Seitenlänge in einem Webbrowser. Die Seiten sollen möglichst weniger Scrollen benötigen, um zu verschiedenen Bereiche zuzugreifen. Die wichtigsten und die meistgebrauchten Elemente sollen im oberen Bereich der Seite positioniert werden.
- Navigation und Orientierung sind von zentraler Bedeutung für ein Webangebot. Die Navigation gibt einen Überblick über den Inhalt oder zeigt, wie eine Webanwendung zu bedienen ist. Die soll gut sichtbar und erkennbar bleiben.

Anhang B: Max Script und HTML/JavaScript Code

MaxScript

```
macroScript ExG category:"Expressive Grafik" buttonText:"Expressive Grafik" tooltip:"Expressive Grafik"
(
rollout ExpressiveGrafik „Expressive Grafik“ width:160 height:600
(
    pickButton AddItem „Add Item“ pos:[9,424] width:70 height:20
    MultiListBox lbx1 „Items to be Rendered“ pos:[8,331] width:145 height:5 items:#()
    progressBar pb1 „ProgressBar“ pos:[10,540] width:140 height:12 color:(color 30 10 190)
    button DelItem „Delete Item“ pos:[82,424] width:70 height:20
    button RenderPart „Render“ pos:[8,556] width:70 height:20
    button btn4 „Cancel“ pos:[81,556] width:70 height:20
    groupBox grp1 „Render Items“ pos:[3,317] width:154 height:269
    button btn5 „Create Rotation Camera“ pos:[9,258] width:144 height:20
    spinner spn1 „“ pos:[13,282] width:36 height:16 range:[1,100,25]
    label lbl1 „Number of Frames“ pos:[56,283] width:93 height:14
    groupBox grp2 „Add Cameras“ pos:[3,204] width:154 height:104
    button btn6 „Create Container“ pos:[7,109] width:144 height:20
    button btn7 „Slice Object Mode“ pos:[8,141] width:144 height:20
    pickButton btn8 „Pick Cont.“ pos:[8,165] width:65 height:23
    pickButton btn9 „Pick Objects“ pos:[77,165] width:75 height:23
    button btn10 „Create Axonom.Camera“ pos:[9,223] width:144 height:20
    groupBox grp3 „Define Items“ pos:[3,90] width:154 height:104
    bitmap bmp3 „Bitmap“ pos:[2,7] width:155 height:75 fileName:"EV.jpg"
    label lbl2 „Set Size (regular)“ pos:[8,458] width:50 height:27
    spinner spn2 „“ pos:[56,469] width:45 height:16 range:[1,1200,1150]
    spinner spn3 „“ pos:[107,469] width:45 height:16 range:[1,800,600]
    label lbl3 „Width“ pos:[59,455] width:32 height:12
    label lbl4 „Height“ pos:[109,455] width:34 height:12
    label lbl7 „Set Size (zoom)“ pos:[8,498] width:50 height:26
    spinner spn6 „“ pos:[56,508] width:45 height:16 range:[1,5000,3000]
    spinner spn7 „“ pos:[107,508] width:45 height:16 range:[1,3000,2200]
    label lbl8 „Width“ pos:[59,494] width:32 height:12
    label lbl9 „Height“ pos:[109,494] width:34 height:12

on AddItem pressed do
(
    if $ != undefined do --see if the user selected anything
    (
        sel = $ as array --get selection array in a variable
        GroupedObj = #()

        for s = 1 to sel.count do --check if there is a group in the selection
        (
            if isGroupMember Selection[s] then (append GroupedObj sel[s]) --add all group children to
```

```
separate array
        )
        else false
    )
    for s = 1 to GroupedObj.count do --for all group children
    (
        pos = finditem sel GroupedObj[s] --find and
        sel = deleteItem sel pos --delete all children from a group from the selection array (keep
only the group name)
    )
    --print GroupedObj
    for i = 1 to sel.count do
        lbx1.items = append lbx1.items (sel [i].name as string) --add all elements from selection to listbox
        --lbx1.items = append lbx1.items (sel [i] as string)
    )
)

on DelItem pressed do
(
    if lbx1.items.count > 0 and lbx1.selection != undefined do --Check if there are any items, and if any selected
    (
        for i in lbx1.selection while lbx1.selection.count != undefined do --when something selected
        (
            lbx1.items = deleteItem lbx1.items (lbx1.selection as array) [1] --delete the selected
elements
        )
    )
)

local current_object = 0 --this is local to the RenderPart macroScript
on RenderPart pressed do
(
    --allObjects = objects as array --collect all objects
    allObjects = lbx1.items --collect all objects names from multiselectbox
    current_object += 1 --increment the counter

    if allObjects.count > 0 then --if anything in the multiselectbox
    (
        for current_object = 1 to allObjects.count do --from the first to the last object do
        (
            hide $* --hide all objects
            --myObj = getNodeByName allObjects[current_object] as string
```

```

name
    myObj = execute („$“ + allObjects[current_object]) --get the object with the specified
    --GroupObj = isGroupHead myObj
    --print GroupObj
    unhide myObj-- then unhide the object

    if myObj.children.count != 0 then --check if the object is a group or has multilevel groups
    (
        fn children obj:myObj = (execute („$“ + (obj.name)+“/*/*/*”)
        select (children())
        children() as array
        theChildren = children obj:myObj as array
        --print theChildren
        unhide theChildren --unhide the hierarchy of children from the myObj
    )

    render camera:$camera01 outputFile:(„Geschoss“ + current_object as string + „.png“)
    vfb:off --render a specific camera, give a file name, and turn the render window on or off
    )

)

messagebox „Rendering is completed! All images are stored under MyDocuments(Eigene Dateien) in folowing
subfolder: 3dsmax\sceneassets\images“
unhide $* --show all objects
current_object = 0 --set variable for next rendering
)

)

createDialog ExpressiveGrafik 160 600
)
)

```

HTML / JavaScript

```

<!DOCTYPE HTML PUBLIC „-//W3C//DTD HTML 4.01//EN“ „http://www.w3.org/TR/html4/strict.dtd“>
<html>
<meta http-equiv=“Content-Type“ content=“text/html; charset=UTF-8“ >
<head>
<title>Expressive Visualisierung</title>

<script src=“js/jquery-1.3.2.min.js“ type=“text/javascript“></script>

<script src=“js/jquery.cluetip.js“ type=“text/javascript“></script>
<script src=“js/jquery.event.drag-1.5.js“ type=“text/javascript“></script>

<script type=“text/javascript“ src=“js/eye.js“></script>
<script type=“text/javascript“ src=“js/utlis.js“></script>
<script type=“text/javascript“ src=“js/zoomimage.js“></script>
<script type=“text/javascript“ src=“js/jquery.threesixty-0.3.0.js“></script>

<script type=“text/javascript“>
$(document).ready(function() {
    //AJAX TOOLTIPS
    $('#tt01').cluetip(); $('#tt02').cluetip(); $('#tt03').cluetip(); $('#tt04').cluetip();
    $('#tt05').cluetip(); $('#tt06').cluetip(); $('#tt07').cluetip(); $('#tt08').cluetip(); $('#tt09').cluetip();

    //ACCORDION MENU
    $('#Navi li:nth-child(even)').hide();
    $('#Navi li:nth-child(odd)').click(
    function(){
        if($(this).next().css(„display“)==“none“ ){
            $(this).parent().find(„li:nth-child(even)“).slideUp(„fast“);
            $(this).parent().find(„li:nth-child(odd)“).removeClass(„selected“);
            $(this).addClass(„selected“).next().slideDown(„fast“);
        }else{
            $(this).next().slideUp(„fast“);
        }
    }
    );

    //BY EXPLODION ADD HOVER IMAGE
    $('#TriggerCB3').hover(
        function() {$$(this).addClass(„Tri“);},
        function() {$$(this).removeClass(„Tri“);}
    );
    $('#TriggerCB4').hover(
        function() {$$(this).addClass(„Tri1“);},
        function() {$$(this).removeClass(„Tri1“);}
    );
    $('#TriggerCB17').hover(
        function() {$$(this).addClass(„Tri2“);},
        function() {$$(this).removeClass(„Tri2“);}
    );
    $('#TriggerCB21').hover(
        function() {$$(this).addClass(„Tri3“);},
        function() {$$(this).removeClass(„Tri3“);}
    );

    //SHOW HIDE LAYERS
    $('#NLayerMain').click( function() {
        $('#Layer2').hide();
        $('#Layer1').show();
        $('#HotspotLage').show();
    });

    $('#NLayerMainA1').click( function() {
        $('#Animation').show();
    });

    $('#NLayerMainA2').click( function() {

```

```

        $($,#Animation').hide();
    });
    $($,#NLayerMainZ1').click( function() {
        $($,#LabelsZoom').show();
    });
    $($,#NLayerMainZ2').click( function() {
        $($,#LabelsZoom').hide();
    });
    $($,#NLayerDiagram').click( function() {
        $($,#HotspotLage').hide();
        $($,#Layer1').hide();
        $($,#Layer2').show();
        $($,#NW').hide();
        $($,#SO').show();
    });
    $($,#NordWest').click( function() {
        $($,#SO').hide();
        $($,#NW').show();
    });
    $($,#SudOst').click( function() {
        $($,#SO').show();
        $($,#NW').hide();
    });
    $($,#NLayerDiagramNW').click( function() {
        $($,#SO').hide();
        $($,#NW').show();
    });
    $($,#NLayerDiagramSO').click( function() {
        $($,#SO').show();
        $($,#NW').hide();
    });
    });

    //ZOOM FLOORS
    $($,a.zoom).zoomimage({
        onLoad: function(){
            $($,#zoom01").threesixty({images:arr, method:'click', ,cycle:'2', ,resetMargin':
        }
    });
    $($,a.zoomCB4').zoomimage({});
    $($,a.zoomCB17').zoomimage({});

    //ZOOM LAYERS
    $($,a.zoomLayer').zoomimage({
        controls: false, // display controls
        caption: false, // display caption
        centered: true,
    });

    //EXPLODE FLORS
    $($,#TriggerCB3') //ELEMENT THAT CALLS THE DRAG FUNCTION
        .bind(,drag',function( event ){
            var TopK = 325;
            //alert(Yoffs);
            if(event.offsetY < TopK){
                //BUILDINGPARTS EXPLODE
                $($ ( this ).css( {top: event.offsetY});
                $($ ( #OGTop' ).css( {top: event.offsetY - TopK});
                $($ ( #OG02' ).css( {top: 3*(event.offsetY - TopK)/4});
                $($ ( #OG03' ).css( {top: (event.offsetY - TopK)/2});
                $($ ( #OG01' ).css( {top: (event.offsetY - TopK)/4});
                //ANIMATION LAYER EXPLODE
                $($ ( #ATop' ).css( {top: event.offsetY - TopK});
                $($ ( #A02' ).css( {top: 3*(event.offsetY - TopK)/4});
                $($ ( #A03' ).css( {top: (event.offsetY - TopK)/2});
            }
        }
    });

```

FUNCTION

FUNCTION

```

        $($, #A01' ).css( {top: (event.offsetY - TopK)/4});
        //LABELS EXPLODE
        $($ ( #t04' ).css( {top: event.offsetY - TopK});
        $($ ( #t03' ).css( {top: 3*(event.offsetY - TopK)/4});
        $($ ( #t02' ).css( {top: (event.offsetY - TopK)/2});
        $($ ( #t01' ).css( {top: (event.offsetY - TopK)/4});
        //LABELS ZOOM EXPLODE
        $($ ( #z02' ).css( {top: 3*(event.offsetY - TopK)/4});
        $($ ( #z03' ).css( {top: (event.offsetY - TopK)/2});
        $($ ( #z04' ).css( {top: (event.offsetY - TopK)/4});
    }
}
else{
    //BUILDINGPARTS STOP EXPLOSION
    $($ ( this ).css( {top: TopK});
    $($ ( #Renderings' ).children(,img').css( {top: 0});
    $($ ( #Animation' ).children(,img').css( {top: 0});
    $($ ( #Labels' ).children(,div').css( {top: 0});
    //LABELS ZOOM EXPLODE STOP EXPLOSION
    $($ ( #z02' ).css( {top: 0});
    $($ ( #z03' ).css( {top: 0});
    $($ ( #z04' ).css( {top: 0});
}
});
    $($,#TriggerCB4')
        .bind(,drag',function( event ){
            var TopK = 441;
            if(event.offsetY < TopK){
                $($ ( #LZCB17' ).hide();
                //BUILDINGPARTS EXPLODE
                $($ ( this ).css( {top: event.offsetY}); //TOP ELEMENT THAT CALLS THE DRAG
                $($ ( #GB4_4' ).css( {top: event.offsetY - TopK});
                $($ ( #GB4_3' ).css( {top: 3*(event.offsetY - TopK)/4});
                $($ ( #GB4_2' ).css( {top: (event.offsetY - TopK)/2});
                $($ ( #GB4_1' ).css( {top: (event.offsetY - TopK)/4});
                //LABELS ZOOM EXPLODE
                $($ ( #z05' ).css( {top: 3*(event.offsetY - TopK)/4});
                $($ ( #z06' ).css( {top: (event.offsetY - TopK)/4});
            }
        }
    });
    $($,#LZCB17').show();
    //BUILDINGPARTS STOP EXPLOSION
    $($ ( this ).css( {top: TopK});
    $($ ( #RenderingsCB4' ).children(,img').css( {top: 0});
    //LABELS ZOOM EXPLODE STOP EXPLOSION
    $($ ( #z05' ).css( {top: 0});
    $($ ( #z06' ).css( {top: 0});
}
});
    $($,#TriggerCB17')
        .bind(,drag',function( event ){
            var TopK = 250;
            if(event.offsetY < TopK){
                //BUILDINGPARTS EXPLODE
                $($ ( this ).css( {top: event.offsetY}); //TOP ELEMENT THAT CALLS THE DRAG
                $($ ( #GB17_2' ).css( {top: event.offsetY - TopK});
                $($ ( #GB17_1' ).css( {top: (event.offsetY - TopK)/2});
                //LABELS ZOOM EXPLODE
                $($ ( #z08' ).css( {top: (event.offsetY - TopK)/2});
                $($ ( #z10' ).css( {top: (event.offsetY - TopK)/2});
            }
        }
    });
    //BUILDINGPARTS STOP EXPLOSION
    $($ ( this ).css( {top: TopK});
}

```

```

    $( '#RenderingsCB17' ).children(,img').css( {top: 0});
    //LABELS ZOOM EXPLODE STOP EXPLOSION
    $(, #z08' ).css( {top: 0});
    $(, #z10' ).css( {top: 0});
    }
});

$(, #TriggerCB21'
    .bind(,drag,function( event ){
        var TopK = 100;
        if(event.offsetY < TopK){
            $(, #CB3' ).hide();
            $(, #GhostingCB3' ).show();
            //BUILDINGPARTS EXPLODE
            $( this ).css( {top: event.offsetY}); //TOP ELEMENT THAT CALLS THE DRAG
        }
    });

FUNCTION
    $(, #GB21_5' ).css( {top: event.offsetY - TopK});
    $(, #GB21_4' ).css( {top: 4*(event.offsetY - TopK)/5});
    $(, #GB21_3' ).css( {top: 3*(event.offsetY - TopK)/5});
    $(, #GB21_2' ).css( {top: 2*(event.offsetY - TopK)/5});
    $(, #GB21_1' ).css( {top: (event.offsetY - TopK)/5});
    }
    else{
        $(, #CB3' ).show();
        $(, #GhostingCB3' ).hide();
        //BUILDINGPARTS STOP EXPLOSION
        $( this ).css( {top: TopK});
        $(, #RenderingsCB21' ).children(,img').css( {top: 0});
    }
});

$(, #dsoGB3_6'
    .bind(,drag,function( event ){
        var Toffs = 0;
        var Yoffs = event.offsetY;
        if(Yoffs < 1){
            //BUILDINGPARTS EXPLODE
            $( this ).css( {top: event.offsetY + Toffs}); //TOP ELEMENT THAT CALLS THE
        }
    });

DRAG FUNCTION
    $(, #dsoGB3_5' ).css( {top: 4*event.offsetY/5 + Toffs});
    $(, #dsoGB3_4' ).css( {top: 3*event.offsetY/5 + Toffs});
    $(, #dsoGB3_3' ).css( {top: 2*event.offsetY/5 + Toffs});
    $(, #dsoGB3_2' ).css( {top: event.offsetY/5 + Toffs});
    }
    else{
        //BUILDINGPARTS STOP EXPLOSION
        $( this ).css( {top: Toffs});
        $(, #RenderingsSO' ).children(,img').css( {top: 0});
    }
});

$(, #dnwGB3_7'
    .bind(,drag,function( event ){
        var Toffs = 0;
        var Yoffs = event.offsetY;
        if(Yoffs < 1){
            //BUILDINGPARTS EXPLODE
            $(, #dnwGB3_6' ).css( {top: event.offsetY + Toffs}); //TOP ELEMENT THAT
        }
    });

CALLS THE DRAG FUNCTION
    $(, #dnwGB3_5' ).css( {top: 4*event.offsetY/5 + Toffs});
    $(, #dnwGB3_4' ).css( {top: 3*event.offsetY/5 + Toffs});
    $(, #dnwGB3_3' ).css( {top: 2*event.offsetY/5 + Toffs});
    $(, #dnwGB3_2' ).css( {top: event.offsetY/5 + Toffs});
    }
    else{
        //BUILDINGPARTS STOP EXPLOSION
        $(, #RenderingsNW' ).children(,img').css( {top: Toffs});
    }
});
});
});

//360 GESCHOSS ROTATION
//
var arr = [];
//loop to add the images in the array
for (var x=2; x<= 25; x++)
    arr.push(,images/zoom1/zoom_ " + x + ,.png");
}); // doc.ready
</script>
<link rel="stylesheet" media="screen" href="css/styles.css" type="text/css" >
<link rel="stylesheet" media="screen" href="css/jquery.cluetip.css" type="text/css" >
<link rel="stylesheet" media="screen" type="text/css" href="css/zoomimage.css" >
</head>
<body>
<div class="Body1">
<div id="Body2">
<div id="Layer1">

<div id="LabelsUmg">
<a id="tt01" style="top: 660px; right: 360px;" href="tt1.htm" rel="tt1.htm" title="The Tower: Company Building 03"></a>
<a id="tt02" style="top: 590px; left: 600px;" href="tt1.htm" rel="tt1.htm" title="Company Building 04"></a>
<a id="tt03" style="top: 440px; right: 580px;" href="tt1.htm" rel="tt1.htm" title="Company Building 21"></a>
<a id="tt04" style="top: 350px; left: 210px;" href="tt1.htm" rel="tt1.htm" title="Company Building 17-20"></a>
<a id="tt05" style="top: 50px; left: 550px;" href="tt1.htm" rel="tt1.htm" title="Donau Kanal"></a>
<a id="tt06" style="top: 320px; right: 130px;" href="tt1.htm" rel="tt1.htm" title="Autobahn"></a>
<a id="tt07" style="top: 150px; left: 180px;" href="tt1.htm" rel="tt1.htm" title="Nachbar Gebauden ??"></a>
<a id="tt08" style="top: 670px; right: 180px;" href="tt1.htm" rel="tt1.htm" title="U-Bahn Station Erdberg"></a>
<a id="tt09" style="top: 640px; left: 280px;" href="tt1.htm" rel="tt1.htm" title="Nachbar Gebauden ??"></a>
</div>
<div id="CB17">
<div id="RenderingsCB17">


<div id="LZCB17"> <a href="images/Outside5_ZOOM.png" class="zoomCB17" id="z08" style="margin-top: 390px;"></a>
<a href="images/Outside4_ZOOM.png" class="zoomCB17" id="z09" style="margin-top: 420px;"></a>
<a href="images/zoom1/zoom_01.png" class="zoomCB17" id="z10" style="margin-top: 340px;"></a>
<a href="images/zoom1/zoom_01.png" class="zoomCB17" id="z11" style="margin-top: 370px;"></a>
</div>
</div>
</div>
<div id="CB21">
<div id="RenderingsCB21">

```




6. Bilderverzeichnis

Abb. 1: Getrennte Darstellungen von Innen- und Aussenraum (Orestad in Kopenhagen von Daniel Liebeskind)	11
[Studio Daniel Liebeskind: http://www.daniel-libeskind.com/projects/show-all/orestad-downtown-master-plan/ , 2009]	
Abb. 2: Interaktive expressive Visualisierung [Zaeva, Biljana: <i>Interaktive Architekturvisualisierung</i>]	12
Abb. 3: Einsatz von Exploded und Ghosted View (Sustainable Workplace von Paul Weston) [Paul Weston Associates: http://www.graphicwest.co.uk/ , 2008]	16
Abb. 4: Exploded View nach Niederauer et al., 2003. Drei explodierte Levels aus dem Spiel Quake III	
[NIEDERAUER, Christopher; HOUSTON, Mike; AGRAWALA, Maneesh; HUMPHREYS, Greg: <i>Non-invasive interactive visualization of dynamic architectural environments</i> . ACM Transactions on Graphics, Volume 22, Issue 3, 2003]	17
Abb. 5: Helferobjekte in eine Exploded View Architekturvisualisierung	
[ISA: Exploded View of 100K House - http://www.flickr.com/photos/postgreen/2233397752/ , 2008]	18
Abb. 6: Exploded View in einer Richtung [Foster + Partners: <i>Spaceport America</i> - http://www.fosterpluspartners.com/Projects/1613/Default.aspx , 2009]	18
Abb. 7: Exploded View in drei Richtungen	
[Integrated Publishing: <i>Exploded view of a typical light-frame house</i> - http://www.tpub.com/content/construction/14043/css/14043_23.htm , 2009]	19
Abb. 8: Gelenkige Exploded View	
[MCGUFFIN M., TANCAU L., BALAKRISHNAN R.: <i>Using deformations for browsing volumetric data</i> . In Proceedings of IEEE Visualization'03; S. 401–408., 2003]	19
Abb. 9: Teilexplosion [LI, Wilmot; AGRAWALA, Maneesh; SALESIN, David: <i>Interactive image-based exploded view diagrams</i> . Graphics Interface, S. 203-212, 2004]	20
Abb. 10: Cut-Away View [Zaeva Biljana: <i>Ägypten</i> , 2001]	20
Abb. 11: Cutout View [Yankee Stadium Visualisation von IOMEDIA: http://newyork.yankees.mlb.com/nyy/ballpark/new_stadium_seat_selector.jsp , 2009]	21
Abb. 12: Breakaway View [Kevin Hulsey Illustration Inc. - http://www.khulsey.com/ , 2009]	21
Abb. 13: Schnittlinien bei Cut-Away View (Center for Arts and Culture in Boston von Daniel Liebeskind)	
[Studio Daniel Liebeskind: http://www.daniel-libeskind.com/projects/show-all/new-center-for-arts-and-culture/ , 2009]	22
Abb. 14: Ghosted View [Beaudaniels illustration - http://beaudaniels.com/Architectural-pages/Gas-company-house.htm , 2009]	22
Abb.15: Ghosted View [Beaudaniels illustration - http://beaudaniels.com/Architectural-pages/Cement-factory.htm , 2009]	23
Abb. 16: Phantom View [Beaudaniels illustration - http://beaudaniels.com/Architectural-pages/Bridge-span.htm , 2009]	23

Abb. 17: Internal Labels [HUANG, Jingshu; BUE, Brian; PATTATH, Avin; EBERT, David S.; THOMAS, Krystal M.: <i>Interactive Illustrative Rendering on Mobile Devices</i> . IEEE Computer Graphics and Applications, vol. 27, no. 3, S. 48-56, 2007]	24
Abb. 18: Externe Labels mit zusätzlicher Legende [ISA: Exploded View of 100K House - http://www.flickr.com/photos/postgreen/2233397752/ , 2008]	24
Abb.19: Verwendung von Linien[HI5: Video Remind Me - http://www.h5.fr/]	25
Abb. 20: Exploded View mit Pfeile und eingezoomte Detailelementen [ITEDO Intelligent Technical Documentation - http://www.itedo.com , 2007]	25
Abb. 21: a) Axonometrie; b) Perspektive [Kevin Hulsey technical illustration - http://www.khulsey.com/ , 2009]	26
Abb. 22: Bleistift Darstellungen [PROVOST, James: Portfolio - http://www.jamesprovost.com , 2009]	27
Abb. 23: Hatching Straffuren [Quelle: PRAUN, E.; HOPPE, H.; WEBB, M.; A. FINKELSTEIN, A.: <i>Real-time hatching</i> . ACM SIGGRAPH 2001 Conference Proceedings, S. 581-586, 2001]	28
Abb. 24: Strippling Straffuren [Quelle: Springer, Maximilian: <i>Non-Photorealistic Rendering</i> , 2007]	28
Abb. 25: Cartoon Rendering [PROVOST, James: Portfolio - http://www.jamesprovost.com , 2009]	28
Abb. 26: Umgebungsrendering (Trutec Building in Seoul von Barkow Libinger Architects) [Archispass Blog - http://www.archispass.org/?p=372 , 7 August, 2006]	29
Abb. 27: Detailstufen in Stadtmodelle [TRAPP, Matthias; GLANDER, Tassilo; BUCHHOLZ, Henrik; DÖLLNER, Jürgen: <i>3D Generalization Lenses for Interactive Focus + Context Visualization of Virtual City Models</i> . 12th International Conference on IEEE Information Visualization, S. 356-361, 2008]	30
Abb. 28: Einsatz von Semantische Schärfentiefe (Insel bei New York) [„governor’s island“ by diller scofidio + renfro - http://www.designboom.com/weblog/cat/9/view/4893/governors-island-by-diller-scofidio-renfro.html , 2008]	30
Abb. 29: Einsatz von Beleuchtung (Visualisierung WU Wien) [Zaha Hadid Architects: WU Wien - http://architecturelab.net/2009/02/03/winners-of-vienna-university-campus-competition-selected/ , 2009]	31
Abb. 30: Einsatz von Farbe (Rendering für das Dongdaemun Design Park) [Zaha Hadid Architects: Dongdaemun Design Park and Plaza Starts Construction - http://www.archinnovations.com/news/new-projects/zaha-hadid-architects-dongdaemun-design-park-plaza-starts-construction/ , 2009]	31
Abb. 31: 2D Darstellung von Exploded View (Olympic Stadium London von Populous)[London 2012 Olympic and Paralympic Games - http://www.london2012.com/venues/olympic-stadium.php , 2009]	33
Abb. 32: Spaceport America von Foster + Partners [Foster + Partners - http://www.fosterpluspartners.com/Projects/1613/Default.aspx , 2009]	34
Abb. 33: Spaceport America von Foster + Partners [Foster + Partners - http://www.fosterpluspartners.com/Projects/1613/Default.aspx , 2009]	34
Abb. 34: Wettbewerbprojekt von Gilles LEFEVRE and Mathieu Badie architects [Wikipedia - http://commons.wikimedia.org/wiki/File:Concours_architecture_tour_gustave_eiffel_2008_2009_projet_dessous_dame_fer_3D_%C3%A9tudiant_gilles_lefevre_mathieu_badie_antipode.jpg , 2009]	34
Abb.35: Nationalbibliothek in Prag von Future Systems a) Cut-Away View; b) Exploded View; c) Eploded View mit Guides [Future Systems: <i>Prague library</i> - http://www.bdonline.co.uk/story.asp?storycode=3082843 , 2007)	35
Abb. 36: Automatischerzeugte Exploded Views von der Spiel Quake III [NIEDERAUER, Christopher; HOUSTON, Mike; AGRAWALA , Maneesh; HUMPHREYS, Greg: <i>Non-invasive interactive visualization of dynamic architectural environments</i> . ACM Transactions on Graphics, Volume 22, Issue 3, 2003]	36
Abb. 37: Automatischerzeugte Exploded Views von 2D Grafiken[LI, Wilmot; AGRAWALA, Maneesh; SALESIN, David: <i>Interactive image-based exploded view diagrams</i> . Graphics Interface, S. 203-212, 2004]	37
Abb. 38: Music Video „Remind Me“[H5 - http://www.h5.fr/ , 2009]	38

Abb. 39: Infographik für Areva [H5 - http://www.h5.fr/ , 2009]	38
Abb. 40: Webseite von McKinstry [McKinstry - http://www.mckinstry.com , 2009]	39
Abb. 41: T-Com Haus Präsentation [Zebworx & think-quicktime: http://www.think-quicktime.com/de/quicktime/interaktiv/t-com-haus/t-com-haus.html , 2005]	40
Abb. 42: Lageplan von TownTown (Wiener Stadtwerke und Soravia: http://www.towntown.at/hp_german/ , 2009]	43
Abb. 43: Lageplan [Zaeva, Biljana]	44
Abb. 44: a) Raumvisualisierung, b) Wegeführung [Zaeva, Biljana: <i>Interaktive Architekturvisualisierung</i>]	45
Abb. 47: Regelgeschoss [Zaeva, Biljana: <i>Interaktive Architekturvisualisierung</i>]	46
Abb. 45: Workflowdiagram (Abschnitt: 1-3) [Zaeva, Biljana]	47
Abb. 46: Workflowdiagram (Abschnitt: 4-5) [Zaeva, Biljana]	48
Abb. 48: Plug In für 3DS Max [Zaeva, Biljana: aus 3ds Max]	49
Abb. 49: Definition von Elementengruppen [Zaeva, Biljana: aus 3ds Max]	51
Abb. 50: Definition von Gebäudeteile [Zaeva, Biljana: aus 3ds Max]	51
Abb. 51: Erzeugte Grafiken von den Rendering Durchlauf [Zaeva, Biljana: <i>Interaktive Architekturvisualisierung</i>]	52
Abb. 52: Interface [Zaeva, Biljana: <i>Interaktive Architekturvisualisierung</i>]	53
Abb. 53: Hauptlayer-System [Zaeva, Biljana: <i>Interaktive Architekturvisualisierung</i>]	54
Abb. 54: Explosion Feature [Zaeva, Biljana: <i>Interaktive Architekturvisualisierung</i>]	54
Abb. 55: Ghosting Feature [Zaeva, Biljana: <i>Interaktive Architekturvisualisierung</i>]	55
Abb. 56: Zoom Feature [Zaeva, Biljana: <i>Interaktive Architekturvisualisierung</i>]	56
Abb. 57: Frames für den Rotation Feature [Zaeva, Biljana: <i>Interaktive Architekturvisualisierung</i>]	56
Abb. 58: Labels [Zaeva, Biljana: <i>Interaktive Architekturvisualisierung</i>]	57
Abb. 59: a) JPEG Grafikformat; b) GIF Grafikformat [Zaeva, Biljana]	64
Abb. 60: Browserstatistiken - Weltweit [W3Schools - http://www.w3schools.com/browsers/browsers_stats.asp , 2009]	65
Abb. 61: Browserstatistiken - Österreich [StatCounter - http://gs.statcounter.com , 2009]	65
Abb.62: a) Interface; b) Direkte Modelmanipulation [Zaeva, Biljana: <i>Interaktive Architekturvisualisierung</i>]	67

7. Literaturverzeichnis

- AGRAWALA, Maneesh; PHAN, Doantam; HEISER, Julie; HAYMAKER, John; KLINGNER, Jeff; HANRAHAN, Pat; TVERSKY, Barbara: *Designing effective step-by-step assembly instructions*. In ACM SIGGRAPH'03 Papers, S. 828-837, 2003
- ALI, Kamran; HARTMANN, Knut; STROTHOTTE, Thomas: *Label Layout for Interactive 3D Illustrations*. Journal of the WSCG, Volume 13, S. 1-8, 2005
- BENDER, Michael: *Computergrafik : ein anwendungsorientiertes Lehrbuch* - Carl Hanser Verlag, 2006. (ISBN 3-446-40434-1)
- BEAUDOUIN-LAFON, Michel: *Instrumental Interaction: An Interaction Model for Designing Post-WIMP User Interfaces*. Proceedings of the SIGCHI conference on Human factors in computing systems, S.446-453, 2000
- COCONU, Liviu; DEUSSEN, Oliver; HEGE, Hans-Christian: *Real-Time Pen-and-Ink Illustration of Landscapes*. Proceedings of the 4th international symposium on Non-photorealistic animation and rendering, S. 27 - 35, 2006
- DIEPSTRATEN, J; WEISKOPF, D; ERTL, T: *Interactive cutaway illustrations*. Computer Graphics Forum, Volume 22, no. 3, S. 523-532, 2003
- DIEPSTRATEN, J; WEISKOPF, D; ERTL, T: *Transparency in interactive technical illustrations*. Computer Graphics Forum, Volume 21, no. 3, S. 317-325, 2002
- HERCZEG, Michael: *Software-Ergonomie : Grundlagen der Mensch-Computer-Kommunikation* - Oldenbourg Wissenschaftsverlag GmbH, 2005. (ISBN 3-486-25052-3)
- HEISER, Julie; PHAN, Doantam; AGRAWALA, Maneesh; TVERSKY, Barbara; HANRAHAN, Pat: *Identification and Validation of Cognitive Design Principles for Automated Generation of Assembly Instructions*. Advanced Visual Interfaces, S. 311-319, 2004
- HOPPE, Hugues (Microsoft Research): *Progressive meshes*. International Conference on Computer Graphics and Interactive Techniques, S. 99-108, 1996
- HUANG, Jingshu; BUE, Brian; PATTATH, Avin; EBERT, David S.; THOMAS, Krystal M.: *Interactive Illustrative Rendering on Mobile Devices*. IEEE Computer Graphics and Applications, vol. 27, no. 3, S. 48-56, 2007
- KALBACH, James: *Handbuch der Webnavigation* - O'Reilly Verlag GmbH & Co. KG, 2008. (ISBN 978-3-89721-865-9)

- KATO, Michio: *Axonometry and New Design of Bauhaus*. Journal for Geometry and Graphics, Volumen 11, S.73-82, 2007
- KOSARA, Robert; MIKSCH, Silvia; HAUSER, Helwig, *Focus+Context Taken Literally*. IEEE Computer Graphics and Applications, vol. 22, no. 1, S. 22-29, 2002
- KRIKKE, Jan: *Axonometry: A matter of Perspective*. IEEE Computer Graphics and Applications, Volume 20, no. 4, S. 7-11, 2000
- LIVNY, Boaz: *Mental ray for Maya, 3ds Max and XS : a 3D artist's guide to rendering* - Wiley Publishing Inc., 2008. (ISBN 978-0-470-00854-6)
- LI, Wilmot; AGRAWALA, Maneesh; CURLESS, Brian; SALESIN, David: *Automated Generation of Interactive 3D Exploded View Diagrams*. ACM Transactions on Graphics, S. 101:1-101:7., 2008
- LI, Wilmot; AGRAWALA, Maneesh; SALESIN, David: *Interactive image-based exploded view diagrams*. Graphics Interface, S. 203-212, 2004
- LI, Wilmot; RITTER, Lincoln; AGRAWALA, Maneesh; CURLESS, Brian; SALESIN, David: *Interactive Cutaway Illustrations of Complex 3D Models*. SIGGRAPH, S. 31:1-31:11, 2007
- MCGUFFIN M., TANCAU L., BALAKRISHNAN R.: *Using deformations for browsing volumetric data*. In Proceedings of IEEE Visualization'03, S. 401-408., 2003
- MOHAMMED, Riaz; KROLL, Ehud: *Automatic generation of exploded view by graph transformation*. Ninth Conference on Artificial Intelligence for Applications, S. 368-374, 1993
- NIEDERAUER, Christopher; HOUSTON, Mike; AGRAWALA, Maneesh; HUMPHREYS, Greg: *Non-invasive interactive visualization of dynamic architectural environments*. ACM Transactions on Graphics, Volume 22, Issue 3, 2003
- NIELSON, Jacob: *Designing Web Usability* - Markt+Technik Verlag, 2001 (ISBN: 3-8272-6206-2)
- NINDL, Gotfried: *Eine Ajax Literaturdatenbank mit ADOWeb*. Diplomarbeit TU Wien, 2008
- PREIßLER, Angelika: *Implementierung eines historischen 3D-Weltatlases auf der Basis von X3D*. Masterarbeit TU Wien, 2008
- PRAUN, E.; HOPPE, H.; WEBB, M.; A. FINKELSTEIN, A.: *Real-time hatching*. ACM SIGGRAPH 2001 Conference Proceedings, S. 581-586, 2001
- SILVA CAETANO, Christian da : *3D-Architektur-Visualisierung : Atmosphäre mit Konzept, Licht und Struktur in 3ds Max* - Heidelberg : mitp, 2008 (ISBN 978-3-8266-5923-2)
- SLUSALLEK, Philipp: *Vision An Architecture for Physically-Based Rendering*. Doktorarbeit Universität Erlangen-Nürnberg, 1995
- SPOOL, J.M.; SCANLON, T.; SCHROEDER, W.; SNYDER, C.; DEANGELO, T.: *Web Site Usability: A Designer's Guide* - Morgan Kaufmann Publ., 1999 (ISBN 1-55860-569-X)
- TRAPP, Matthias; GLANDER, Tassilo; BUCHHOLZ, Henrik; DÖLLNER, Jürgen: *3D Generalization Lenses for Interactive Focus + Context Visualization of Virtual City Models*. 12th International Conference on IEEE Information Visualization, S. 356-361, 2008

TUFTE, Edward R.: *Visual Explanations: Images and Quantities, Evidence and Narrative* - Graphics Press USA, 1997 (ISBN-10: 0961392126)

TUFTE, Edward R.: *Beautiful Evidence* - Graphics Press USA, 2006 (ISBN-10: 0961392177)

VIOLA, Ivan: *Importance-Driven Expressive Visualization*. PhD Thesis TU Wien, 2005

VIOLA, Ivan; GRÖLLER, Meister: *Smart Visibility in Visualization*. In Proceedings of EG Workshop on Computational Aesthetics Computational Aesthetics in Graphics, Visualization and Imaging, S. 209-216, 2005

VOLLICK, Ian; VOGEL, Daniel; AGRAWALA, Maneesh; HERTZMANN, Aaron: *Specifying Label Layout Styles by Example*. Symposium on User Interface Software and Technology, S. 221 - 230 , 2007

Webseiten:

Autodesk - <http://www.autodesk.de/>, 2009

Berlin Hauptbahnhof, Audioguide, Virtueller Rundgang - http://www.hbf-berlin.de/site/berlin__hauptbahnhof/de/bahnhofsrundgaenge/audiofuehrung/audiofuehrung.html, 2008

Detail: Praxis+Trends, Internet - http://www.detail.de/thema_trends-onlinemarketing_145_De.htm, 2009

Die freie Enzyklopädie - www.wikipedia.org, 2009

DONATH, Prof. Dr.-Ing. D.: *Bildbearbeitung: Vektorgraphik*. Fachzeitschrift Technische Dokumentation - <http://www.doku.net/artikel/vektorgraf.htm>, 2000

GARRETT, Jesse James: *Ajax: A New Approach to Web Applications*. Adaptive Path LLC - <http://www.adaptivepath.com/ideas/essays/archives/000385.php>, 2005

German Chapter der Usability Professionals' Association e.V. - <http://germanupa.de/>, 2009

H5: *Infografiken* - <http://www.h5.fr/>, 2009

Kevin Hulse technical illustration - <http://www.khulsey.com/>, 2009

Knight Digital Media Center, The Online Journalism Review: *Animated infographics and online storytelling: Words from the wise* - <http://www.ojr.org/ojr/stories/070523ruel/>, 2007

LUCKHARDT, Heinz-Dirk: *Labels als Mittel der Informationsarchitektur* - <http://is.uni-sb.de/studium/handbuch/labeling.html>, 2003

Meyers Konversations-Lexikon - <http://www.peter-hug.ch/lexikon>, 1988

MÜNZ, Stefan: *SELFHTML* - http://www.validome.org/doc/HTML_ge/index.htm, 2005

Online-Magazin Internet-fuer-Architekten: <http://internet-fuer-architekten.de/>, 2009

REYNOLDS, Craig: *Stylized Depiction in Computer Graphics: Non-Photorealistic, Painterly and, Toon Rendering* - <http://www.red3d.com/cwr/npr/>, 2003

TOWNTOWN - <http://www.towntown.info/>

Thomson Reuters: *remerinc's New Website for McKinstry... a Moving Experience* - <http://www.reuters.com/article/pressRelease/idUS95910+06-May-2009+PRN20090506>, 2009

Web3D Consortium: <http://www.web3d.org/x3d/>, 2009

Wienenergie: *Fernkälte* - <http://www.wienenergie.at/we/ep/contentView.do?contentType=1001&programId=12413&channelId=-22449&contentId=16866>

Software & Support Verlag GmbH: *Interview mit Frank Puscher, Thema Usability* - <https://www.entwickler.com/zonen/portale/psecom,id,99,news,37003,.html>

Zebworx & think-quicktime: <http://www.think-quicktime.com/de/quicktime/interaktiv/t-com-haus/t-com-haus.html>, 2005