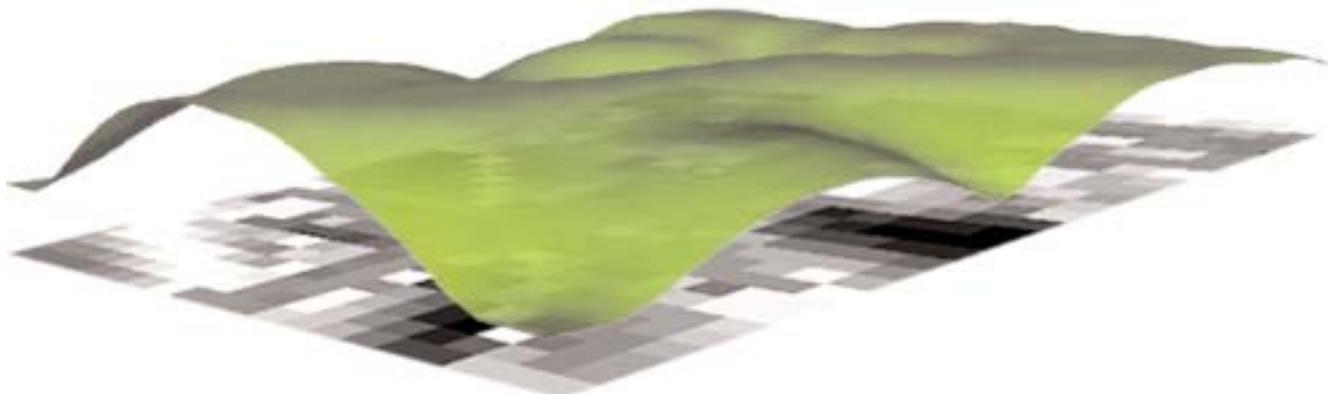


# LAUT.02



## Diplomarbeit über "bewegungsorientierte Raumfindung"

ausgeführt zum Zwecke der Erlangung des akademischen Grades eines Diplom-Ingenieurs.

unter der Leitung von:  
Univ.Prof. Dipl.-Ing. arch. Manfred Wolff-Plottegg  
E252 - Institut für Gebäudelehre und Entwerfen

eingereicht an der Technischen Universität Wien - Fakultät für Architektur und Raumplanung.

von:  
**Christoph Falkner** [94.27624] und **Thomas Grasl** [95.26476]  
beide wohnhaft in A-1070 Wien | Kirchengasse 38/9

Wien, im Oktober 2002



Christoph Falkner

Thomas Grasl

Im Laufe unseres Studiums haben wir uns immer mehr vom klassischen Entwurfsdenken, bei dem ein Architekt eine schöne Idee hat, diese zu Papier bringt, plant und schließlich seine stilistischen Gedanken baut, distanziert und nach neuen Planungsmethoden und anderen Planungsparametern gesucht. Durch die neuen Medien und medialen Vernetzungen, speziell aber durch die computerunterstützten Methoden in der Architektur wurde es möglich, stil- und interpretationsfreier zu denken und zu arbeiten. Vor wenigen Monaten noch war es unsere Absicht den architektur-romantischen Gedanken "Häuser zum Laufen" zu bringen zu unserem Thema zu machen. Mit den ersten Schritten, wie dem "Raumfilm" oder dem "mover" wurde dieser Gedanke als Basis für das Simulationsprogramm "ve.loc.it.y" aufgenommen. "ve.loc.i.ty" entwickelte sich zu einem Simulationsprogramm mit dem sich schließlich ein auf Bewegung basierender Raum generieren lässt. Diese Raumfindung läuft in einer Schleife, so lange bis ein stabiler Zustand erreicht ist und ohne unser stilistisches Zutun baubar wird. Das Resultat unserer Arbeit ist dieses automatisiertes Programm, welches Simulationsmodelle für alle Maßstäbe, vom Einraumhaus bis zu städtebaulichen Landschaften, liefert. Diese sind funktions- und bis zu einer gewissen Weise interpretationsfrei. Anhand von Beispielen wollen wir unsere Vorstellung einer Weiterführung des Programmes präsentieren.

Für die Unterstützung in den letzten Jahren möchte ich mich bei meinen Eltern, Großeltern und Tante Christa bedanken.

Christoph Falkner

Mutter, Onkel Hans.

Ich danke Euch für Eure Unterstützung.

Thomas Grasl

01   Einleitung	Absichtserklärung	0001
	Vorwort	0002
	Startschuß	0004
02   der Mover	Verräumlichung von Bewegung	0005
	GBLmover - live	0006
	mover - Fassade   Raumfilm	0008
03   ve.loc.i.ty	Allgemeines	0010
	A-Star	0011
	Aufbau von ve.lo.c.i.ty	0012
	Anwendung   Benutzeroberfläche	0015
	Funktionsschemen	0017
	Attraktoren   Funktionsschemen	0020
	erste Anwendung	0022
	Stapeln weiterer Ebenen	0027
	mittige Simulationsebene	0031
04   Anwendungen	Allgemeines	0033
	[S] das Wolkenhaus	0034
	[S] die Wohnkapsel	0037
	[M] das Pavillion	0042
	[L] die Baulücke	0054
	das Pyramidenproblem	0056
	[XL] Europaplatz	0061
05   Ergänzungen	Glossar	0065
	Quellenverzeichnis	0071

### HINWEIS:

Zum zusammenhängenderen Verständnis gibt es im Glossarteil ergänzende Beschreibungen zu oft verwendeten Begriffen und deren Bedeutung bezogen auf diese Diplomarbeit.

Der vorliegenden Diplomarbeit gehen zwei Entwurfsprogramme voraus, bei denen wir vieles angedacht haben, was uns beim vorliegenden Projekt als Basis diente.



Abb.01: "Stadtfluidität" - Modell der negativen Bewegungsdarstellung



Abb.02 | 03: Manipulation als Entwurfsansatz.



Abb.04: Wegführungsmodell in der "dichten Lücke"



Abb.05: Parameter für die "dichte Lücke"

Details zum Projekt "Shopping in der dichten Lücke" findet man im Web unter:  
<http://turn.to/swapplus>

## Stadtfluidität

Zum einen ist dies "Stadtfluidität" aus dem Jahre 1999 am Institut für Wohnbau: Damals experimentierten wir mit Beschleunigung und Verzögerungen. In modellhaften Darstellungen wird diese negativ dargestellt, d.h ein Auto das sich einer Kreuzung nähert, bildet eine ansteigende Kurve. Je mehr beschleunigt wird, desto niedriger wird die Kurve. Eine Flußmündung, welche stellenweise Wirbel bildet wird zu einem massiven Raum - der konstant, langsamer fließende Fluß hingegen bildet eine Senke. Unser damaliges Experimentierumfeld war die Wienflußmündung bei der Urania.

## Shopping in der dichten Lücke

Das zweite Projekt ist "shopping in der dichte Lücke" aus dem Jahre 2001 am Institut für Hochbau 1: Hier haben wir erstmals ein Architektur generierendes Programm entwickelt, das in einem dreidimensionalen Raum zunächst ein Netzwerk von schnellstmöglichen Verbindungen erstellt, diese später bündelt und so optimierte Siedlungsflächen für Konsumattraktoren schafft. Parameter dafür waren Dichte, Zeit, Useranzahl, Geld und Zeit. Diese Anwendung wurde für U-Bahnstationen ausgelegt, ein Beispielprojekt war der Knoten Stephansplatz in Wien.



## 01 | Verräumlichung von Bewegung - "the mover"

Die ursprüngliche Idee von "the mover" war es ein Video in ein Raster zu teilen und jedes Rasterfeld auf seine Bewegungsintensität hin zu überprüfen. Das Kriterium für die automatisierte Erkennung für Bewegung ist in diesem Fall die Intensität der Farbänderung zwischen zwei benachbarten Frames. In den ersten Scripts folgte dann eine Neuordnung der Felder nach der Gesamtbewegungsintensität, d.h. über die ganze Länge des Videos gesehen. Wobei oben links die meiste und unten rechts die geringste Bewegung stattfand.

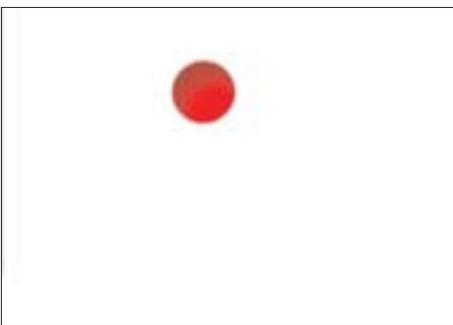


Abb.06: Film: Eine rote Kugel rollt von links nach rechts über einen weißen Fläche

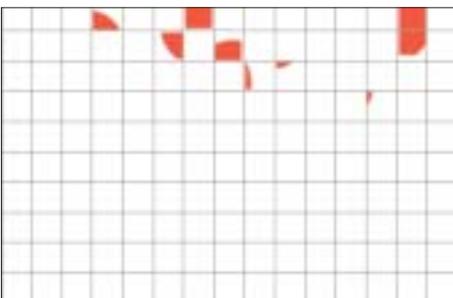


Abb.07: Neuordnung der Rasterfelder nach Bewegungsintensität von links oben nach rechts unten.

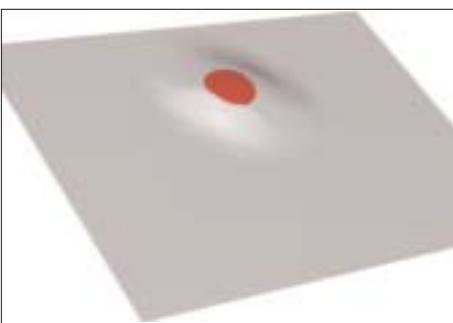


Abb.08: Der Raumfilm der roten Kugel

### PSEUDOCODE mover1

```

get movie
Loop number of frames times
    examine the differences
    add up the difference for each field in array
next
sort array
Loop number of frames times
    reorder fields according to array
    write frame to reordered movie
next
save reordered movie

```

In weiterer Folge wurde für jedes Frame die Bewegung der jeweiligen Rasterfelder in eine Textdatei geschrieben. Diese Daten wurden dann in ein 3D Programm eingelesen und auf ein Netz aufgetragen, so dass die Bewegungsintensität direkt proportional mit der Lage des Netzknotens entlang der z-Achse ist. So bekommt der zweidimensionale Film eine dritte Dimension und es entstehen Bewegungsräume. Zuletzt wurde noch das ursprüngliche Video auf die nunmehr animierte Netzfläche projiziert und von einem verschobenen Blickwinkel aus gesehen gerendert. Der Raumfilm entsteht.



Abb.09: Film einer bewegten Situation



Abb.10: Rasterung und Neuordnung...

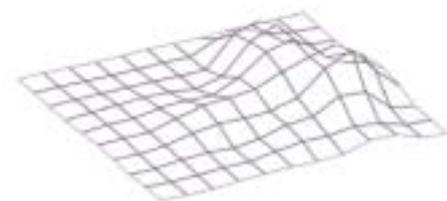


Abb.11: Übertragung als Netz im 3dStudio



Abb.12: Mapping des Originalfilms auf das bewegte Netz.

#### PSEUDOCODE mover2

```

get movie
Loop number of frames times
    examine the differences
    write differences to text file
next

```

#### PSEUDOCODE maxscript

```

get text file
create mesh with appropriate size
do loop
    loop number of vertexes times
        read next value
        move vertex along z-axis by value * multiplier
    next
    move to next frame
until End Of File

```

### "GBLmover" live

Durch eine Modifizierung des "mover" scripts ist es möglich die Rasterung eines Filmes live durchzuführen. Der Film, welcher von einer Webcam aufgezeichnet wird, durchläuft den Mover und wird über einen Bildschirm oder Beamer dem Akteur der Bewegung präsentiert. Ein erstes Beispiel dieser visuellen Interaktion ist der GBLmover, bei dem das Livebild des Ganges vor dem Institut für Gebäudelehre durch den Mover manipuliert und wieder auf den Gang projiziert wurde. Bei jeder Art von Bewegung kommt es zu einer neuen Ordnung des Bildes. Beruhigt sich die Bewegung, so kommt es auch zu einem ruhigem Livebild.

Der GBL-mover ist auch live im Internet.  
Zu finden unter <http://www.gbl.tuwien.ac.at>



Abb.13: Architekturhighway



Abb.14: Movermonitor am Institut für GBL&E

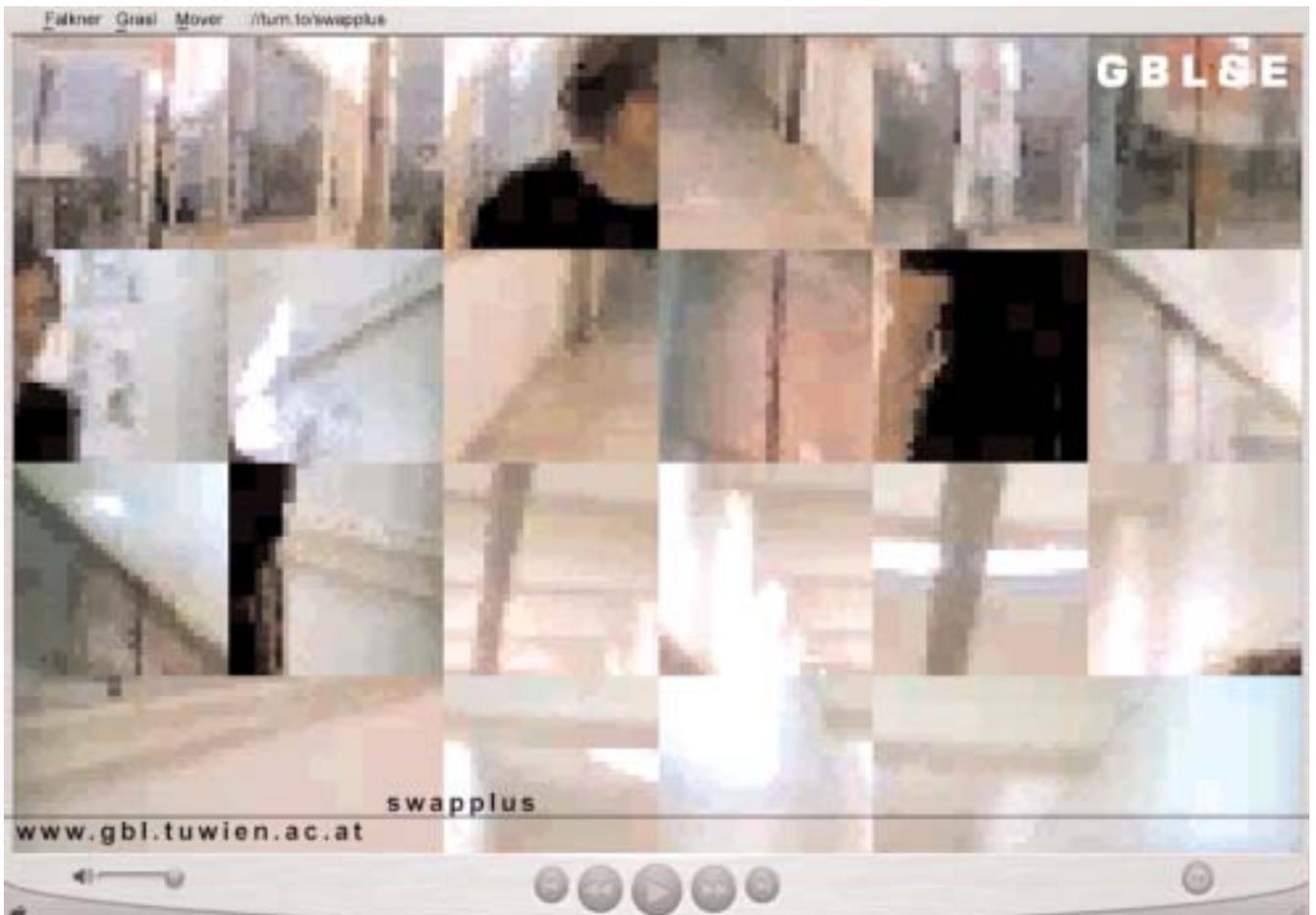


Abb.15: Moverbild



Abb.16: Moverfassade am Westbahnhof

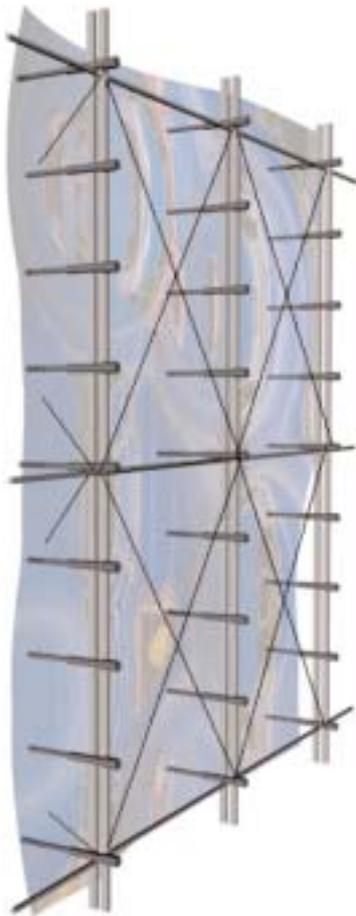


Abb.17: Konstruktion der Moverfassade mit den Steuerelementen.

### Die "Mover"-Fassade

Eine bewegliche Fassade als visueller Aufbruch zwischen Innen und Außen.

Von einem vertikalen Konstruktionsgitter aus wird über gesteuerte Teleskopstangen eine elastische Membranhaut punktuell um den Wert der Bewegungsdifferenz des Livefilmes bewegt. Die Möglichkeiten der Projektion sind zum einen klassisch über Datenbeamer, oder in naher Zukunft über FOLED-Stoffe. [Flexible Organic Light Emitting Device]

Livefilme aus Räumen mit entsprechendem Bewegungsaufkommen werden schließlich auf die Haut projiziert. Durch die Übertragung des Films auf die Konstruktion entsteht ein differenziertes Bild des Gebäudes und eine neue Wahrnehmung von Innen und Aussen, Wand oder Film.



Abb.18: Westbahnhof, hinter dem Raumfilm

### Der Raum hinter dem "Raumfilm"

Die Bewegung im Raum ist direkter Stoff für die Liveprojektion und jeder Benutzer ist Akteur und durch seine Bewegung auch Manipulator der "Moverfassade". So wird die reale Raumordnung durch "the mover" neu geordnet und erhält laufend eine andere Prägung. Im Sinne der Wahrnehmung kommt es zu einer subjektiven Irritation und neuen "Raumbildern". Ähnlich wie bei Peter Weibels "Wand von Lascaux"<sup>1</sup> ist der Nutzer interner Beobachter und gleichzeitig simulierter Manipulator der aus dem Raum ausbricht.

<sup>1</sup> | "Die Wand von Lascaux" Ein Beitrag von Peter Weibel zur Ausstellung "Zeichenbau / real virtualities" im Wiener Künstlerhaus 1999.

## Von der Bewegungssimulation zum generierten Raum

In den nächsten Schritten geht es um die simulierte Bewegung auf einem leeren Feld. Diese wird durch überlagernde Wegführungen von Agenten, welche von einem A.star-Algorithmus gesteuert werden, gebildet. Dieser Prozess wiederholt sich mehrmals und baut jedesmal auf die Spuren der vorhergegangenen Agenten auf. Später kommt ein dem "mover"-Script ähnliches Maxscript in Zusammenhang mit dem 3d-Studio als Raumgenerator wieder zum Tragen.

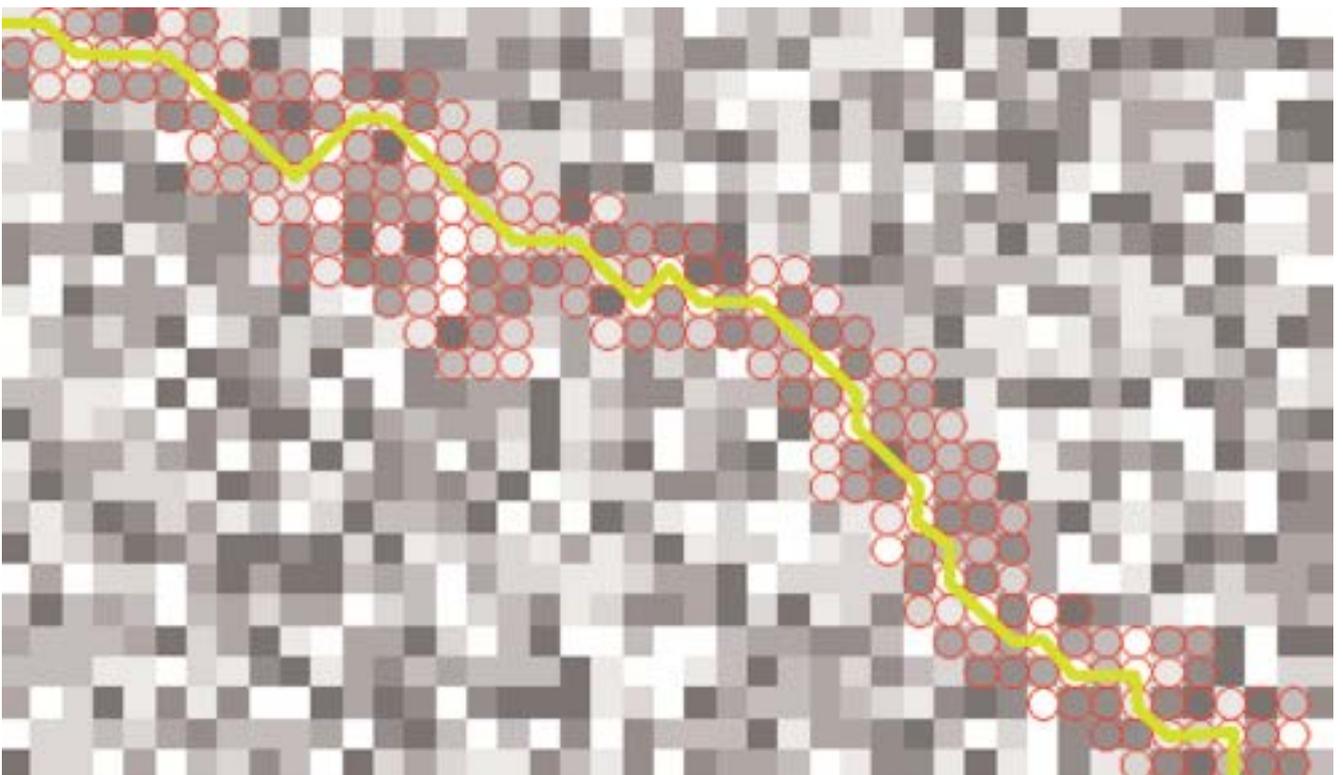


Abb. 19: Berechnungsbild des einfachsten Weges durch den A.star-Algorithmus. Bei diesem Beispiel wurden die Kosten (Graustufe) der einzelnen Felder zufällig bestimmt. Die roten Kreise kennzeichnen die von A.star untersuchten Felder.

## A-Star Algorithmus [A\*]



Abb.20: A\*-Algorithmus by James Macgill - Center for Computational Geography, Leeds.

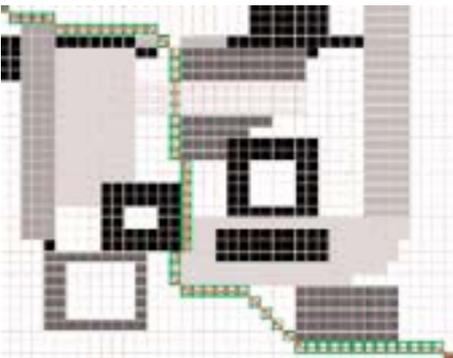


Abb.21: IDDF Pathfinding by Bryan Stout

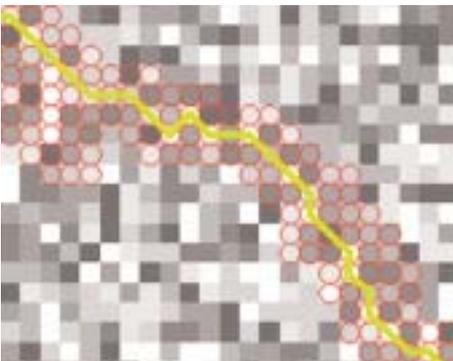


Abb.22: A\* by laut als Basis für ve.loc.i.ty

Dieser Search-Algorithmus ist eine Kombination aus dem Dijkstra Algorithmus und dem best-first Algorithmus. Er wird sehr gerne in der Videospiele-Industrie verwendet, um den mehr oder weniger intelligenten, autonomen Charakteren die Fähigkeit einer realistischen Wegfindung zu geben. Dabei wird in erster Linie nicht nach dem kürzesten, sondern nach dem kostengünstigsten Weg, dem des geringsten Widerstandes, gesucht. Im Fall von ve.loc.i.ty ergibt sich der Widerstand aus einer Kombination der vorhandenen Raumhöhe und dem Gefälle des Untergrunds, wobei die Agenten hohe Räume mit ebenem Boden bevorzugen.

### PSEUDOCODE A\* by Bryan Stout<sup>2</sup>

```
priorityqueue Open
list Closed
AStarSearch
    s.g = 0 // s is the start node
    s.h = GoalDistEstimate( s )
    s.f = s.g + s.h
    s.parent = null
    push s on Open
    while Open is not empty
        pop node n from Open // n has the lowest f
        if n is a goal node
            construct path
            return success
        for each successor n' of n
            newg = n.g + cost(n,n')
            if n' is in Open or Closed,
                and n'.g <= newg
                skip
            n'.parent = n
            n'.g = newg
            n'.h = GoalDistEstimate( n' )
            n'.f = n'.g + n'.h
            if n' is in Closed
                remove it from Closed
            if n' is not yet in Open
                push n' on Open
        push n onto Closed
    return failure // if no path found
```

<sup>2</sup> | Internetquelle: Bryan Stout  
[http://www.gamasutra.com/features/19990212/sm\\_01.htm](http://www.gamasutra.com/features/19990212/sm_01.htm)

Das Programm ve.loc.i.ty dient der Simulation von Agentenströmen. Die Agenten bewegen sich, in einer Umgebung, die aus einer Autocad-Datei eingelesen wurde, von einem zufällig gewählten Eingang A zu einem ebenso zufällig gewählten Ausgang B. Diese Ein- und Ausgänge werden in der Autocad-Datei definiert, wobei man die Gruppen und die Häufigkeiten der Benutzung definieren kann. Somit ist es ein gewichteter und auf eine Gruppe beschränkter Zufallsprozess, der den Weg des Agenten bestimmt. Auf ihrem Weg von A nach B können die Agenten von gewichteten Attraktoren angezogen und wieder weitergeleitet werden. Die primäre Wegfindung wird vom weiter oben beschriebenen A\*-Algorithmus übernommen, auf sekundärer Ebene arbeiten Kollision - Steuerungsverhalten, wie sie von Craig Reynolds<sup>3</sup> beschrieben werden, daran, dass die Agenten einander ausweichen, auch wenn sie dafür den vorher ermittelten Pfad kurzfristig verlassen müssen.

<sup>3</sup> | Internetquelle: Beschreibung des Vermeidungsverhalten von Craig Reynolds [http://www.red3d.com]

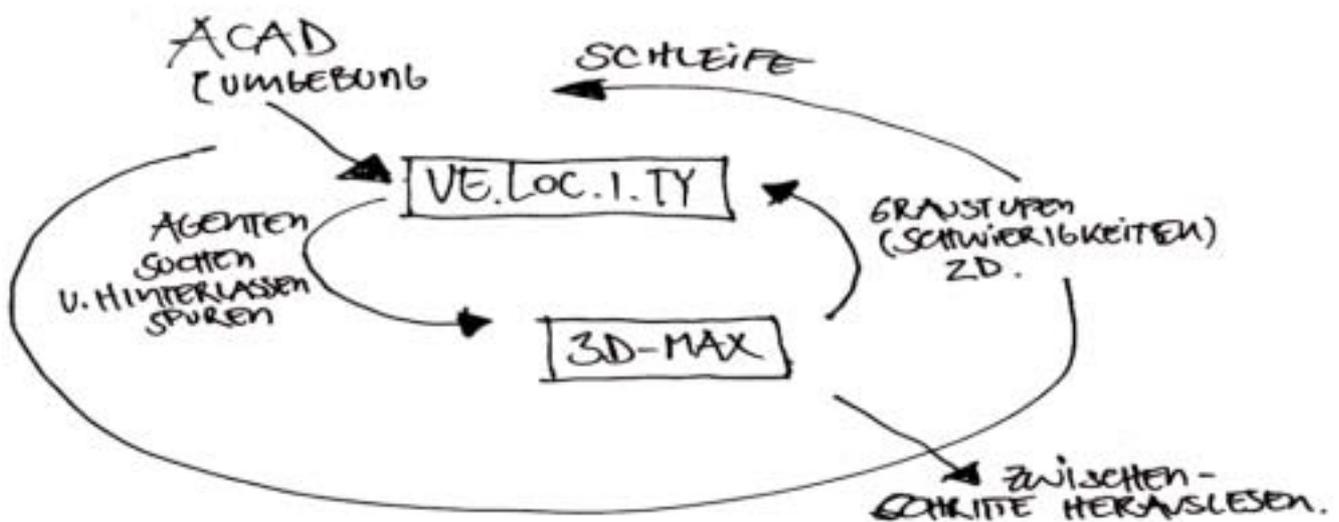


Abb.23: Struktur des velocity-tools

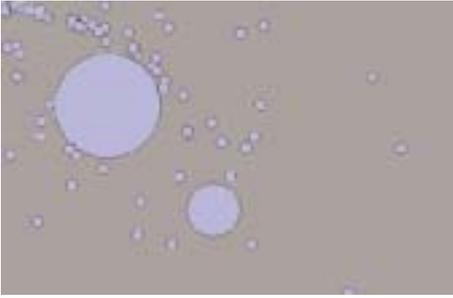


Abb.24: Velocity Generation 001  
Die Agenten laufen von rechts unten nach links oben, die großen Kreise dienen als Platzhalter für zukünftige Agenten

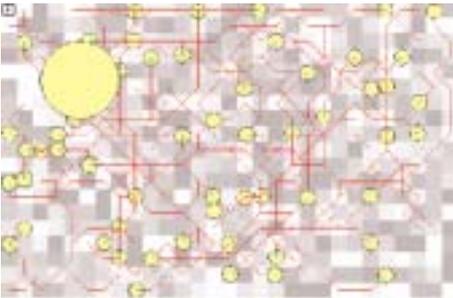


Abb.25: Velocity Generation 002  
Die Agenten markieren ihre Wege und sind mit dem Antikollisionsschutz gerüstet

Die Agenten hinterlassen auf den von ihnen betretenen Feldern eine digitale Spur, aus der in späterer Folge die Bewegungsintensität herausgelesen werden kann. Die gesammelten Daten eines Durchlaufs werden über eine Automatisierungsschnittstelle an 3dsmax übermittelt. 3dsmax liest die Daten der einzelnen Felder aus und überträgt die Werte auf die z-Achse der entsprechenden Knotenpunkte eines Netzes. Retourniert werden die neu ermittelten Kosten der Felder, damit die Agenten die Fläche mit den neuen Parametern erneut begehen können. Es beginnt der nächste Durchlauf.

Dies ist ein rekursiver Prozess. Jede Situation wird durch statische und variable Parameter definiert. Nach jedem Durchlauf ändert sich der variable Teil des Parametersatzes, wodurch ein neuerlicher Durchlauf nötig wird. Die Abbruchbedingung überprüft nach jedem Durchlauf die Gesamtzahl der Änderungen, wenn ein vorgegebener Wert unterschritten, wird hört die Schleife auf. Ziel dieser Rekursivität ist es, eine Situation zu erreichen, die als Optimal bezeichnet werden kann. Dabei ist klar, dass je mehr Parameter eine Situation hat, desto wahrscheinlicher ist es, dass kein oder nur ein lokales Optimum gefunden wird.

### Simulation - Wirklichkeit [design-time / run-time]

In der klassischen Planung [design-time] gibt es eine Entwurfs- und Planungsphase welche alle bautechnisch relevanten Details berücksichtigt und im Bau des Gebäudes resultiert. Die ganze Abwicklung läuft auf einer festgelegten Zeitschiene und ist als aufbauende, additive Methode zu sehen. Die Benutzung des fertigen Gebäudes ist dann die run-time.

Beim vorliegenden Projekt ist diese zeitliche Trennung nicht so einfach, es kommt vielmehr zu einem Wechsel zwischen design-time und run-time (=Simulation).

Die Phase der Parameterfestlegung entspricht einer Art architektonischer design-time. In dieser Zeit wird das Programm entwickelt und auf die Simulation vorbereitet. Die Parameter entsprechen den bautechnischen Details und Bestimmungen, wie z.B. Rampenneigungen, Raumhöhen oder Steigungsverhältnissen. Fehler (design time errors) sind in dieser Situation nicht immer unmittelbar ersichtlich.

Die Folgen dieser Fehler werden erst mit dem Start der Simulation, der run-time, erkennbar und führen zu einer Unterbrechung dieser. Die run-time bezieht sich auf die Ausführungsphase der Simulation. Fehler (run-time-errors) treten, zwar unausweichlich, aber erst unter gewissen Rahmenbedingungen auf. Programmiertechnisch wird dies durch eine ungültige Variablenzuordnung verursacht (z.B. eine Division durch Null), aus architektonischer Sicht, ist dies eine Türöffnung ohne anschließendem Raum oder eben eine zu steil angelegte Rampe. Kommt es zu so einem Laufzeitfehler, so wird eine weitere Nullzeit eingeschaltet und der Prozess durchläuft ein weiteres Mal die design-time. Eine gründliche Testphase kann hier Aushilfe verschaffen.

Ein weiterer Zwischenschritt liegt zwischen Simulation und Realisierung, welcher eine "händische" Verfeinerung von Details beinhaltet, z.B. Anschlüsse der gestapelten Ebenen. Erst im Anschluß ist der baubare Zustand erreicht.

Das baubare Resultat verfügt über eine offene Raumstruktur und lässt ein relativ freie Nutzung zu, kommt es im gebauten Zustand zu gewünschten Änderungen oder Erweiterungen, kann man, sofern es sich um eine flexible Konstruktion handelt, eine weitere design-time einschalten und mit den geänderten Parametern eine neue Simulation starten.

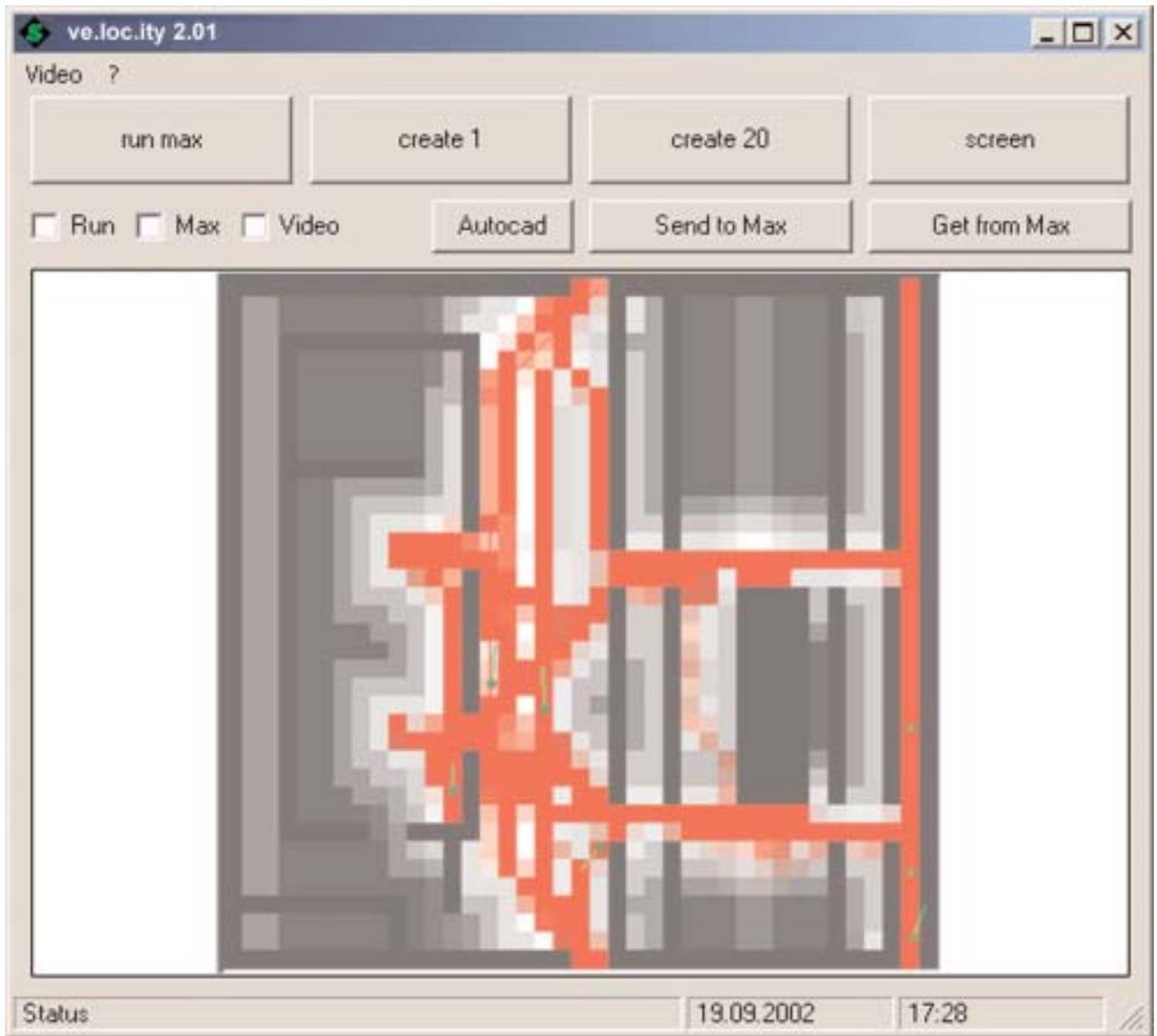


Abb. 26: Die Benutzeroberfläche von ve.loc.i.ty - Version 2.01

**HINWEIS:**

Die aktuellste Version des ve.loc.i.ty-Programmes mit dazu gehörendem Hilfetext ist auf der beiliegenden CD-rom zu finden.

ve.loc.i.ty wurde mit Visual Basic programmiert und funktioniert daher nur auf PCs unter Windows ab der Version 2000. Autocad 2000+, 3D-Studio max 4+ und velocity.dll müssen installiert sein. Weiters ist es nötig 3d-Studio max als OLE-Server zu registrieren, und die entsprechende MaxScript-Datei auszuführen.

Anforderungen an die zu importierende Autocad Datei:

Das Simulationsfeld muss durch ein Rechteck, welches auf dem Layer "simulation" liegt, markiert sein und vom Ursprungspunkt ausgehen. Wände und bestehende Begrenzungen des Feldes sollen auf dem Layer "nogo" liegen und ebenfalls aus Rechtecken aufgebaut sein.

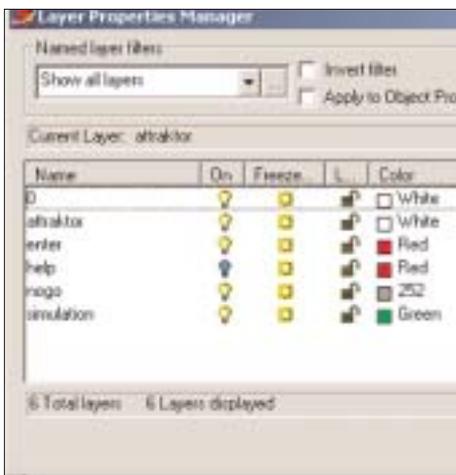


Abb.27: Layerstruktur für die aufbereiteten Umgebungszeichnungen.

Eingänge werden durch Rechtecke, welche am Layer "enter" liegen, definiert. Ein- und Ausgänge können durch die Farbgebung in verschiedene Gruppen geteilt werden. Der Ausgang, den sich der Agent sucht, gehört immer der gleichen Gruppe an wie der von ihm benützte Eingang. Die Eigenschaft "thickness" bestimmt wieviel Agenten durchschnittlich pro Minute einen Eingang benützen.

Sämtliche Attraktoren müssen am Layer "attraktor" liegen. Hier bestimmt die Eigenschaft "thickness" des Rechtecks wieviel Prozent der Agenten diesen Attraktor aufsuchen.

## Verkehrsfläche und Attraktoren

Die Beziehung zwischen Verkehrsfläche und Attraktoren ist eine gegenseitige. Um eine Optimierung der Flächenzuordnungen zu finden, werden verschiedene Funktionsschemen und die damit verbundenen Dichtebildungen in Zusammenhang mit den Ergebnissen von ve.loc.i.ty untersucht.

### lineares Funktionsschema [ Modell 1 ]

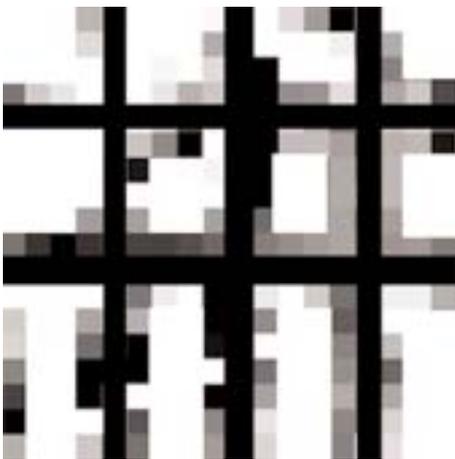


Abb.28: lineares Funktionsschema



Abb.29: Beispiel zu Modell 1:  
Einkaufsstraße, Campton Street, London.

Die Wegführung ist an einer Achse ausgerichtet und an eine Ebene gebunden. Ansiedlungen von Attraktoren finden ausschließlich links und rechts des Weges statt, in den wenigsten Fällen kommt es zu konzentrierten Dichtepunkten. Im Falle einer Einkaufsstraße, welche als lineare Erchließungsachse zu sehen ist, sind dies mehrstöckige Warenhäuser, welche mehrere Agenten anziehen und bezogen auf die Grundfläche, eine höhere Dichte aufweisen. Dieses System ist durch die Rasterplanung im Städtebau entstanden und meist in urbanes Gebiet eingebettet.

Ein weiteres Beispiele für lineare Funktionsschemen in der Architektur sind Gangerschließungen in Wohnungs- und Bürobauten. In den meisten Fällen kommt es hier zu langen Wegen, sprich viel Verkehrsfläche, und geringer Nutzungsdichte.

Im Bezug auf ve.loc.i.ty gilt diese Modell als nicht kompatibel, weil die räumlichen Entwicklungsmöglichkeiten zu gering sind.



Abb.30: flächiges Funktionsschema



Abb.31: Beispiel zu Modell 2:  
Einkaufszentrum, Los Angeles

Im Vergleich zu Modell 1 treten beim flächigen Funktionsschema die klaren Erschließungsachsen in den Hintergrund. Als ein Beispiel kann man Shoppingzentren sehen. Diese, meist geschlossenen Systeme, funktionieren durch eine hohe Attraktorendichte, wobei große Attraktoren einen Teil der Verkehrsfläche übernehmen, kleinere hingegen in Form von Verkaufsständen über die Haupteerschließungsfläche gestreut sind. Die Grenzen zwischen Erschließung und Attraktor verschwimmen. Wegen des relativ großen Maßstabes werden diese Zentren meist aus dem urbanen Kontext gerissen auf grüner Wiese errichtet.

Ein zweites, maßstäblich etwas kleineres Beispiel sind Ausstellungshallen, wo es meistens eine Streuung von Attraktoren in Form von Ausstellungsobjekten gibt und so die Wegführung in einer Netzwerkstruktur endet. Durch die freie Wahl des Weges ist auch eine höhere Agentendichte bei gleichem Nutzungskomfort ("level of service") möglich.

Je nach Absicht und Einsatz des *ve.loc.i.ty*-Tools ist dieses Modell in Betracht zu ziehen. Soll das Ergebnis auf eine Fläche beschränkt bleiben, z.B. für städtische Plätze oder architektonische Landschaften, kann ein flächiges Funktionsschema mit netzwerkartiger Wegführung durchaus interessant sein. Diese Systeme funktionieren meistens nur über zwei Geschoße, da die Treppen und Lifte unattraktiv sind und meistens Engpässe für die Agentenströme darstellen.



Abb.32: räumliches Funktionsschema



Abb.33: Beispiel zu Modell 3:  
optimierte Oberflächennutzung in der internationalen Raumstation ISS.

Dieses Modell übernimmt die Vernetzung von Wegen aus dem flächigen Schema von Modell 2 und überträgt das Netzwerk in die dritte Dimension. Es bedarf daher einer Struktur, die topographisch ausgerichtet ist und mehrere Anschlüsse bzw. räumliche Entwicklungsmöglichkeiten ermöglicht. Die Grundflächen übernehmen durch ihre Neigungen somit auch die vertikale Erschließung, Die Übergänge zwischen Boden und Wänden verfließen, es kommt zu einer Oberflächenerweiterung. Die Trennung zwischen Verkehrsfläche und Attraktoren wird aufgehoben und es kommt zu einer neuen Art von Benutzeroberfläche, welche eine höhere Dichte ermöglicht und gleichzeitig die Bewegungsfreiheit der Benutzer gewährleistet. Lästige, attraktionslose Engpässe wie Treppenaufgänge fallen weg, da es durch die Vielfalt der Wege individuelle Nutzungsmöglichkeiten gibt.

An diesem Modell sollen sich die weiteren Generationen von *ve.loc.i.ty* orientieren, speziell wenn es um den Ausbau in der dritte Dimension geht.

## Allgemeines zu den Attraktoren

In der Simulation zieht ein Attraktor über einen gewissen Zeitraum Agenten an, wobei dieser Zeitraum und die Anzahl der Agenten variabel sind. Die Anordnung der Attraktoren, genauer gesagt der Funktionskerne, kann am leeren Simulationsfeld passieren. Der optimalen Ausnutzung des Raumes halber ist es empfehlenswert, die Platzierung an den wenig frequentierten Orten während des Simulationsablaufes vorzunehmen. Diese simulierten Punkte sind eigentlich nur maßstablose Platzhalter, deren eigentliche Größe vom Simulationsmaßstab und der Größe des Objektes abhängt.

## der Funktionskern

Als Funktionskern versteht man das Zentrum eines Attraktors, den eigentlichen Leistungsbringer, z.B. ein Herd, Tisch oder die Getränkebar im Attraktor Restaurant. Dieses Funktionszentrum kann im Falle von Automaten oder Schließfächern auch ein Attraktor sein, weil es in diesen Fällen keinen unmittelbar zugeschriebenen Raum gibt. Wenn man so will, ist der Funktionskern der reduzierte Attraktor. Sieht man den Funktionskern als den substantiellen Teil eines Attraktors, gelten auch neue Anordnungsregeln. Man kann von einem additiven System, bei dem die Funktionen meistens den Raum definieren, Abstand nehmen und mit offenen Raumgefügen mit flexiblen Funktionspunkten experimentieren.



Abb.34: von Raumdefinitionen freigestellte Funktionskerne für eine Wohnung.



Abb.35: Zwiebelmodell eines Attraktors im klassischen Sinn.

Das Herz ist der oben erwähnte Funktionskern, in dem sich die Dienstleistung des Attraktors abspielt (z.B. die Leinwand im Kino, oder das Sofa im Wohnzimmer).

Die zweite Schicht ist der an den Funktionskern gebundene Raum, für den Benutzer der eigentliche Konsumationsraum. Im Kino sind dies die Sitzplätze, in einem Restaurant kann dies der Gastraum sein.

Als drittes gibt es einen Expansionsraum in die Verkehrsfläche, welche nur zu Spitzenzeiten an den Attraktor gebunden ist. Sollte es zu einer Behinderung der Verkehrsfläche kommen, und so der level of service überschritten werden, hat die Verkehrsfläche Priorität und die Sättigung des Attraktors ist erreicht. Vergleichbares Beispiel ist der Schanigarten vor einem Cafe, der den Gehsteig soweit mitbenutzt, solange ein fließender Verkehr möglich bleibt.

das Zwiebelmodell in einer Wohnung:

In einer Wohnung sind die benutzten Räume genau nach ihren Funktionskernen getrennt und definiert. Dem Raum "Küche" sind zum Beispiel Herd, Kühlschrank, Anrichteflächen und Essbereich als Funktionskerne zugeordnet. Durch diese Trennungen in Zimmer kommt es zu einer großen Verkehrsfläche, die durch die geringe Agentenzahl sehr viel Fläche beansprucht. Kommt es zu einer höheren Benutzeranzahl (z.B. Besuch) so expandiert oder übersiedelt z.B. der Essbereich ins Wohnzimmer. Versuche, die starren Zuordnungen und Raumtrennungen aufzubrechen, ergeben einen eindeutig flexibleren Umgang mit Funktionskernen und der zur Verfügung stehenden Fläche.

Beim Zwiebelmodell wird durch die von der Funktion geprägten Räume eine freie Rauminterpretation und Nutzung gehemmt. Trotzdem ist dieses Modell von "Funktionsräumen" in der klassischen Planung tief verankert.

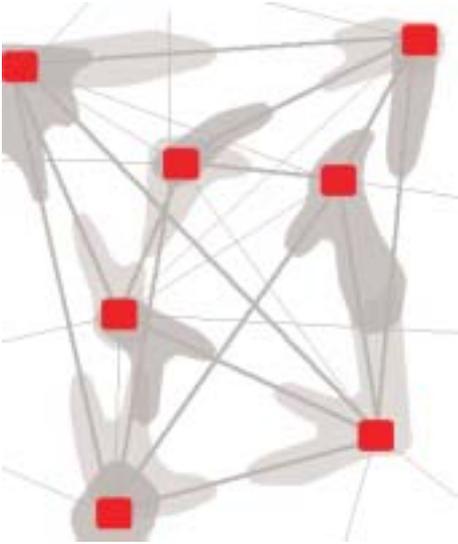


Abb.36: Funktionspunkte die zusammenwachsen.

Um eine flexiblere Anwendung der Attraktoren zu erzielen, und dadurch ein offenes Raumgefüge beizubehalten, wird das oben erwähnte Zwiebelmodell aufgelöst. Genauer gesagt werden nur die Funktionskerne, ohne "benutzten Raum" und "Expansionsfläche" nach den Simulationsergebnissen an die optimalen Punkte platziert. Der benutzte Raum und die gebundenen Flächen werden nicht mehr vordefiniert, sondern sollen sich frei entwickeln, das heißt, dass der Benutzer, der einen Funktionskern benutzt, den Konsumationsraum durch seinen weiteren Weg zeitweilig beansprucht. Werden diese Flächen bzw. Wege von mehreren Agenten genutzt, kann es zu einer Verschmelzung von Funktionskernen kommen.

Wieder ein Wohnbeispiel: Der tägliche Weg vom Bett ins Bad entwickelt sich zu einem Badebett, das als Attraktor raumunabhängig ist.

Im öffentlichen Raum entsteht vielleicht eine "Wurstsemmel"-Achse zwischen Fleischhauer und Bäckerei. Diese Raumachsen werden durch das Benutzerverhalten laufend neu geprägt und ermöglichen einen freien Umgang mit dem offenen Raumgefüge.

Die erste Anwendung des ve.loc.i.ty-Tools basiert auf einem überschaubaren Simulationsfeld mit den Abmessungen von 10m x 14 m. Es werden vier Zugänge vorgegeben. Die Agentendichte liegt bei maximal 40. Nach dem Einlesen des Grundrisses beginnt die Simulation.

In diesem Stadium des Programmes sind die Agentenspuren noch temporär und bauen sich nach einiger Zeit wieder ab. So kommt es zu einer Bewegung der generierten Fläche. Die dazugehörigen Filme der Entwicklung findet man auf der beiliegenden CD.

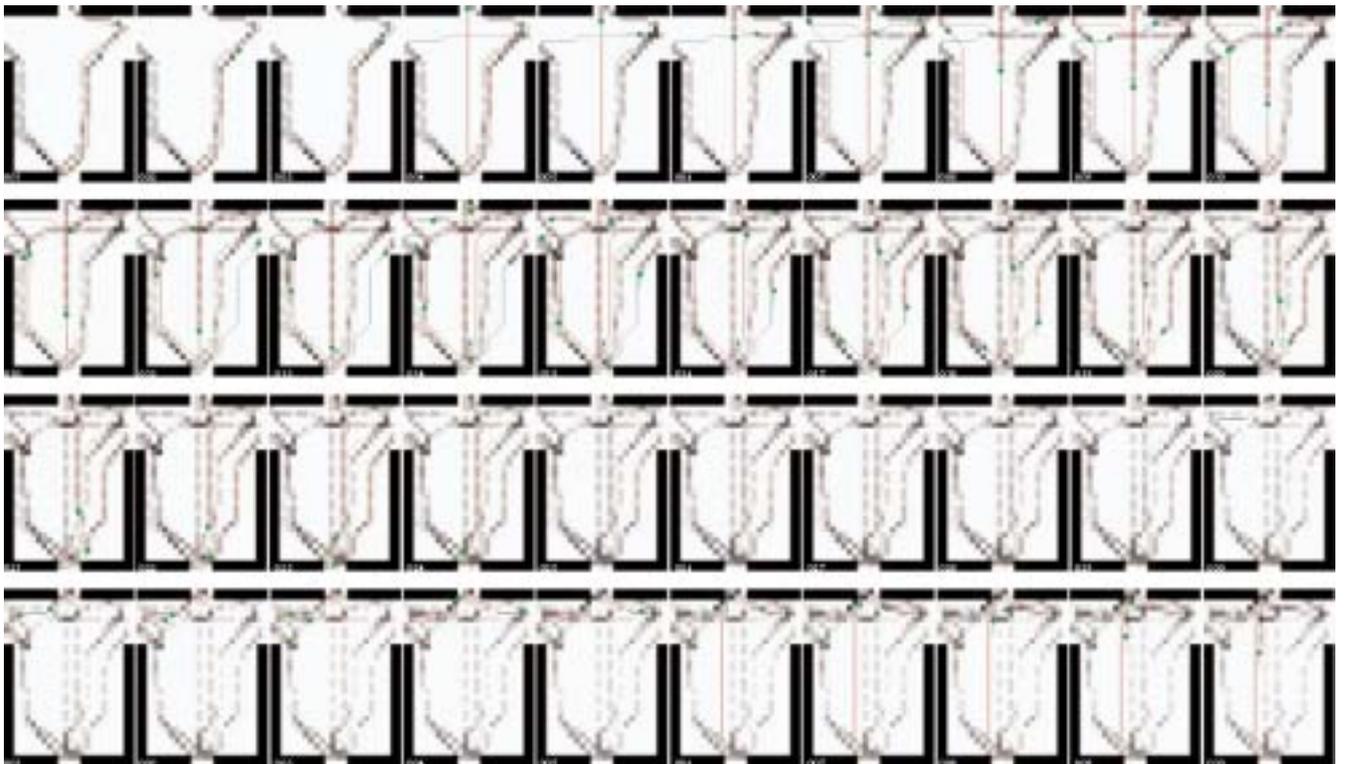


Abb.37: erste Momentaufnahmen der Agentenspuren zwischen den vier Eingängen.

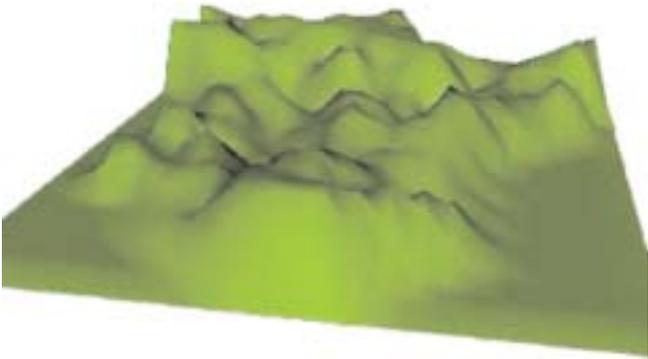


Abb.38: Momentaufnahme des ersten Films des generierten, bewegten Morphs.



Abb.39: Momentaufnahme: Nach abgebauter Überhöhung werden die Übergänge fließender.

### Entwicklungsschritt nach dem ersten Simulationsversuch

Es gibt eine Anzahl an Interpretationsmöglichkeiten, der vom `ve.loc.i.ty`-Tool gewonnenen Daten. Einer der wesentlichsten Unterschiede betrifft die Lebensdauer der von den Agenten gezogenen Datenspuren. So kann man diese Spuren entweder langsam, über die Dauer einer vorgeschriebenen Zeitspanne, verblassen lassen, oder ihnen Beständigkeit während der gesamten Dauer der Simulation geben.

Aber vor allem die von `maxscript` vorgenommene Umlegung der Daten auf das eigentliche Mesh-Objekt erlaubt eine Vielzahl an Variationen und Kombinationen. Hierzu muss man nur das entsprechende Script vor dem Datenaustausch in `3dsmax` laden und ausführen.

Es ist zum Beispiel möglich, den vom Agenten beanspruchten Raum zu modulieren. Dieser Raum kann zurzeit die Form eines Paraboloids oder Pyramide annehmen. Es sind aber auch andere Formen vorstellbar, wobei man natürlich auch die Parameter der einzelnen Formen einstellen kann. So kann man die Pyramide auf eine maximale Steigung von 10% beschränken, um eine rollstuhlge-rechte Ebene zu schaffen. Auch könnte man die Krümmung des Paraboloids auf einen bestimmten Bereich beschränken, um Materialeigenschaften in die Simulation einfließen zu lassen.

Ein weiterer in maxscript einzustellender Faktor betrifft die Auslegung der Simulationsebene. Die absolute Auslegung sieht für jeden Simulationsdurchlauf eine fiktive, horizontale Ebene, auf der sich die Simulation abspielt, und auf die sich die vorgenommenen Änderungen beziehen. Eine andere Variante ist es, die Simulation relativ auf die im vorhergehenden Durchlauf entstandene Ebene zu beziehen. Dadurch stehen die Ebenen in direkter Beziehung zueinander, nicht genutzte Bereiche verschmelzen mit der unteren Ebene, es entstehen keine Resträume.



Abb.40: Beanspruchter Raum aus Paraboloid

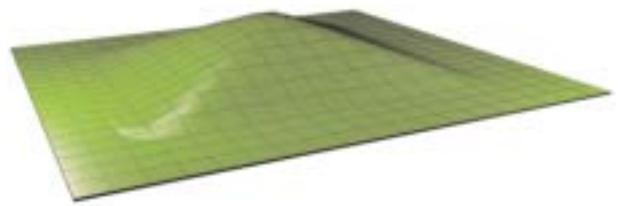


Abb.41: Beanspruchter Raum aus Pyramiden.

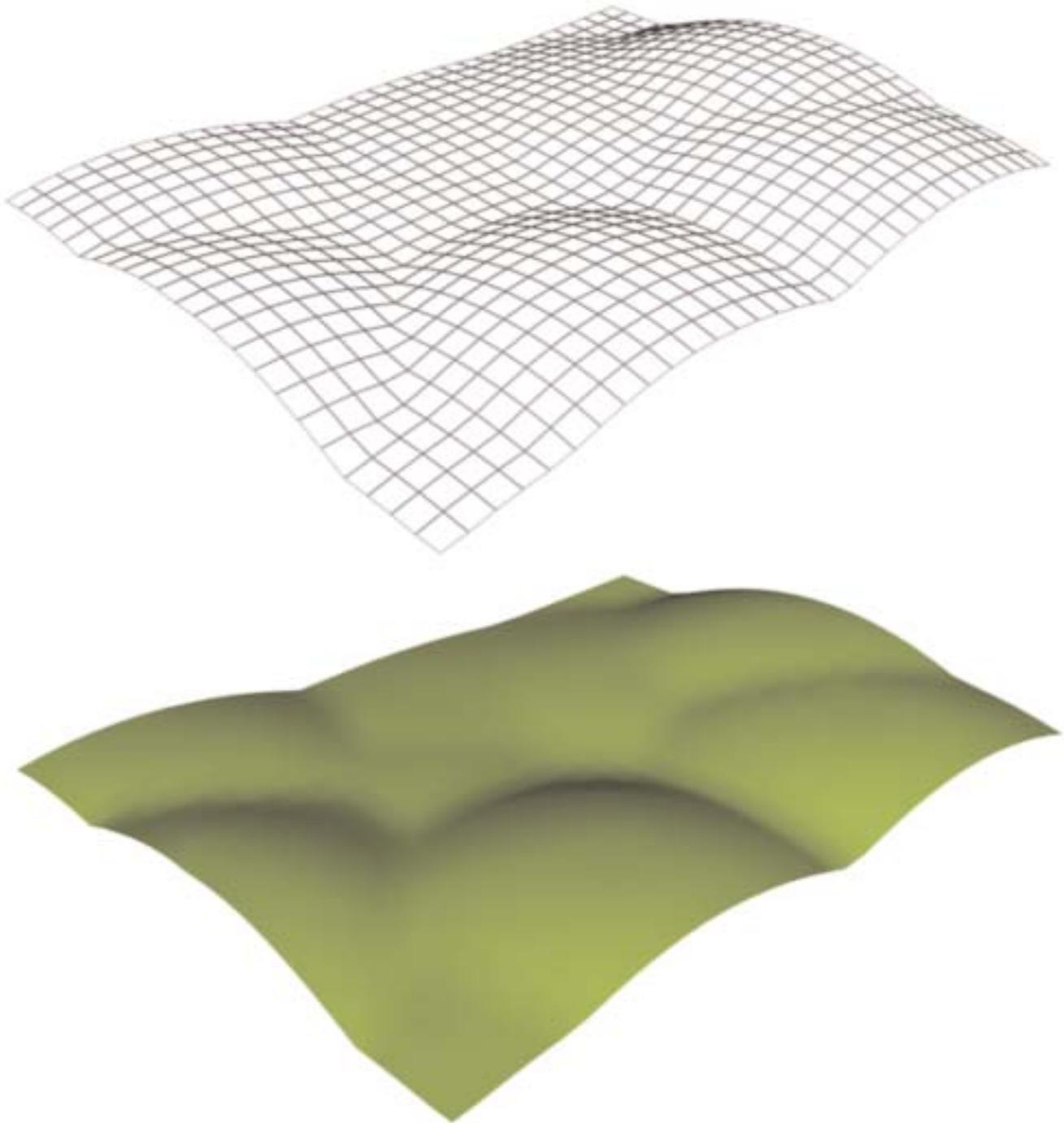


Abb.42: Auf begehbare Neigungen optimierte Fläche.

## Stapeln weiterer Ebenen

Im abgebildeten Beispiel gilt die gleich Ausgangslage wie auf Seite 15. Der einzige Unterschied ist, dass die Agentenspuren nicht mehr temporär sind.

### Ebene 01:

Agentenströme zwischen den vier Zugängen. Zur Annäherung an den stabiler Zustand bedarf es eines mehrmaligen Durchlaufs, ehe weitere Schritte getätigt werden. Diese weiteren Schritte sind das Einsetzen der Funktionskerne für die Attraktoren (grüne Rechtecke), wodurch es zu einer Manipulation der entstandenen Figur kommt. Die Wege der Agenten von, zum oder über den Attraktor stabilisieren sich.

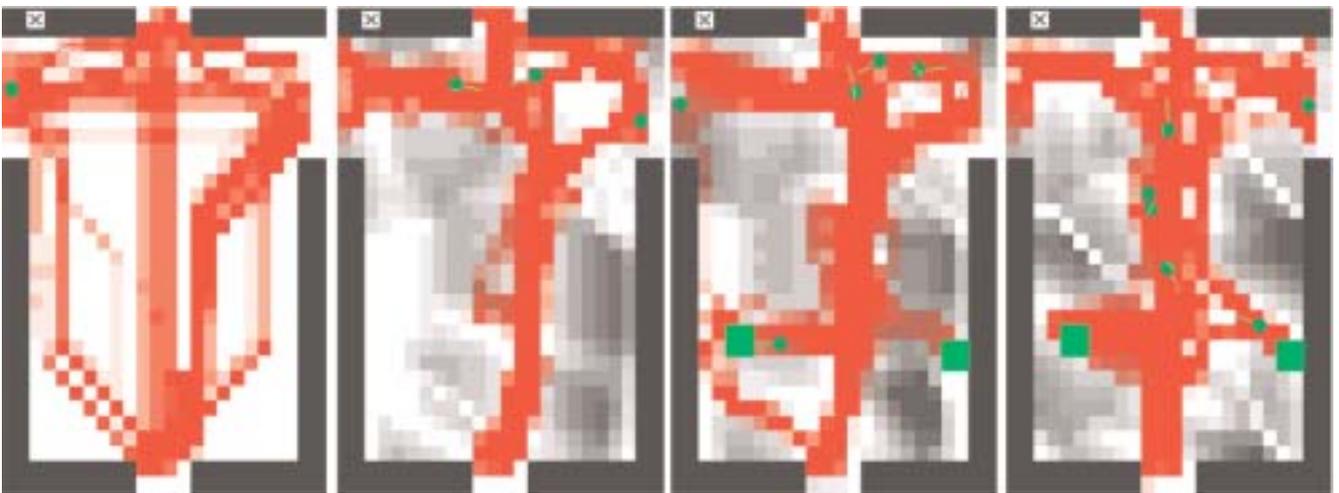


Abb.43: die ersten vier Durchgänge der Agentenströme

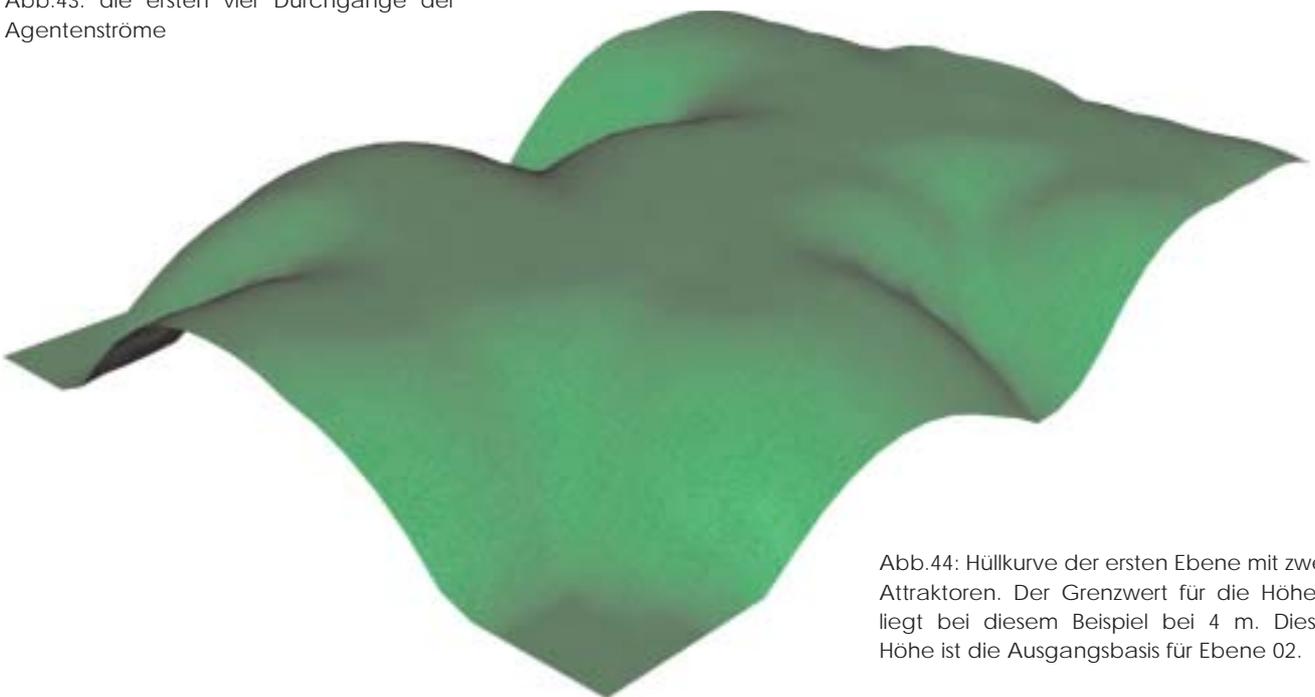


Abb.44: Hüllkurve der ersten Ebene mit zwei Attraktoren. Der Grenzwert für die Höhen liegt bei diesem Beispiel bei 4 m. Diese Höhe ist die Ausgangsbasis für Ebene 02.

### Ebene 02:

Die entstandene Ebene wird vom 3d Studio in die zweidimensionale ve.loc.i.ty-Oberfläche zurückgerechnet und als Simulationsbasis für Ebene02 genutzt. Neue Zugänge (grüne Eckpunkte der linken Figur in Abb.45) von der Ebene 01 definiert, welche die Ebene 02 erschließbar machen. Nach den Stabilisierungsdurchläufen werden weitere Attraktoren gesetzt...

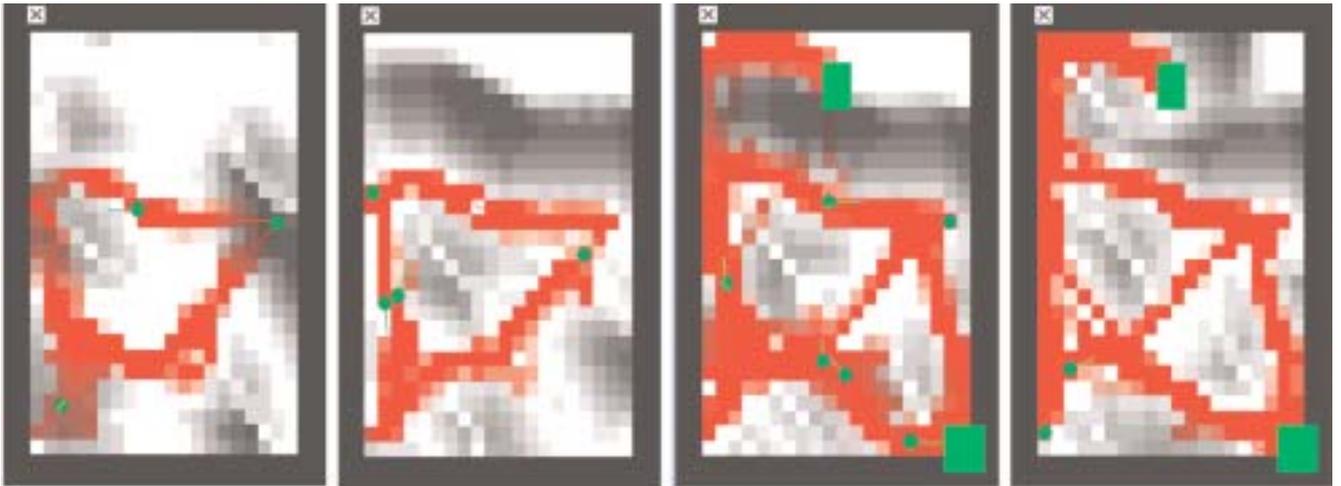


Abb.45: weitere Durchläufe für Ebene 02

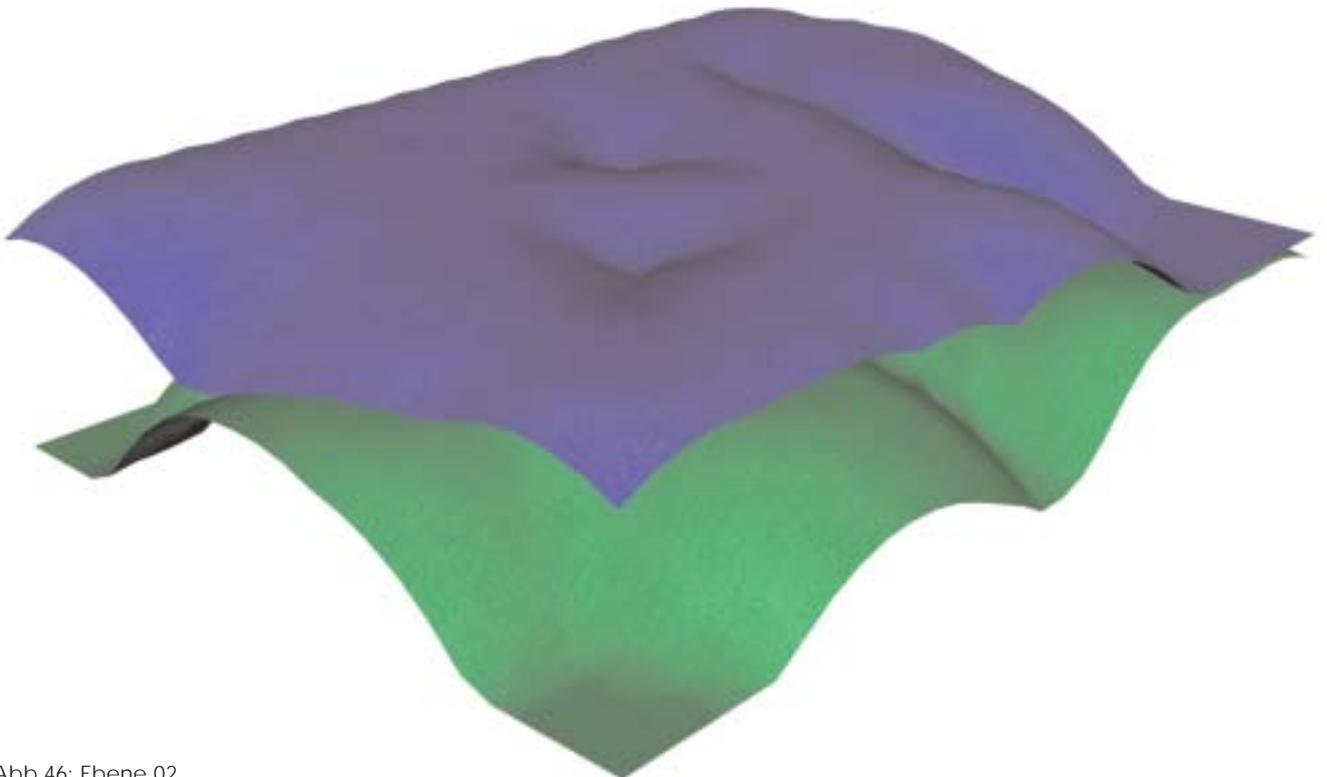


Abb.46: Ebene 02

### Ebene 03:

Diese und alle weiteren Ebenen entstehen nach dem selben Prinzip wie Ebene 02. [ sh. oben ]

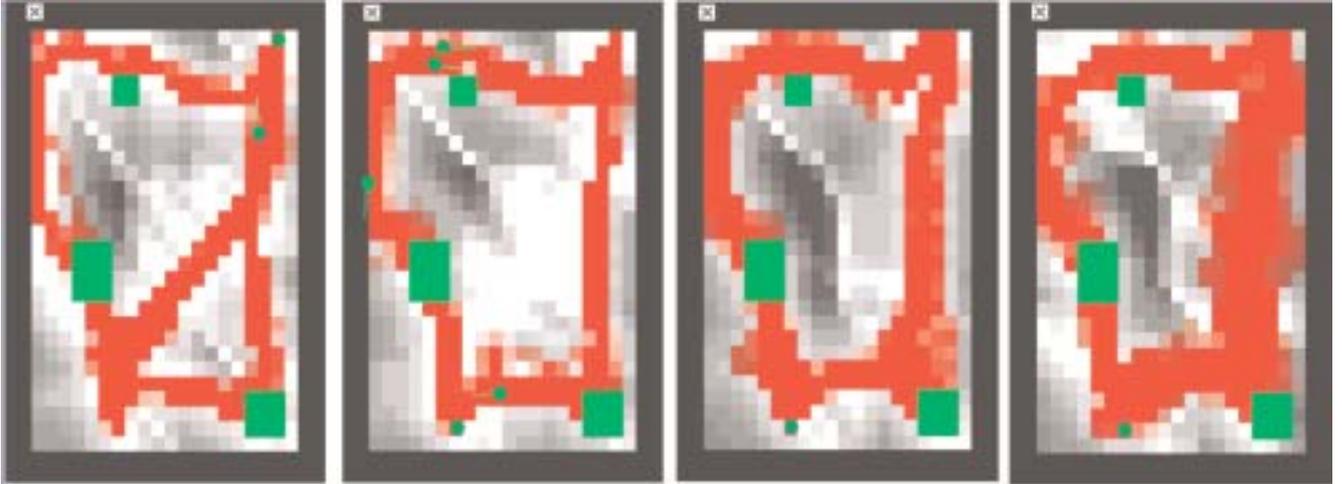


Abb.47: Durchläufe für Ebene 03

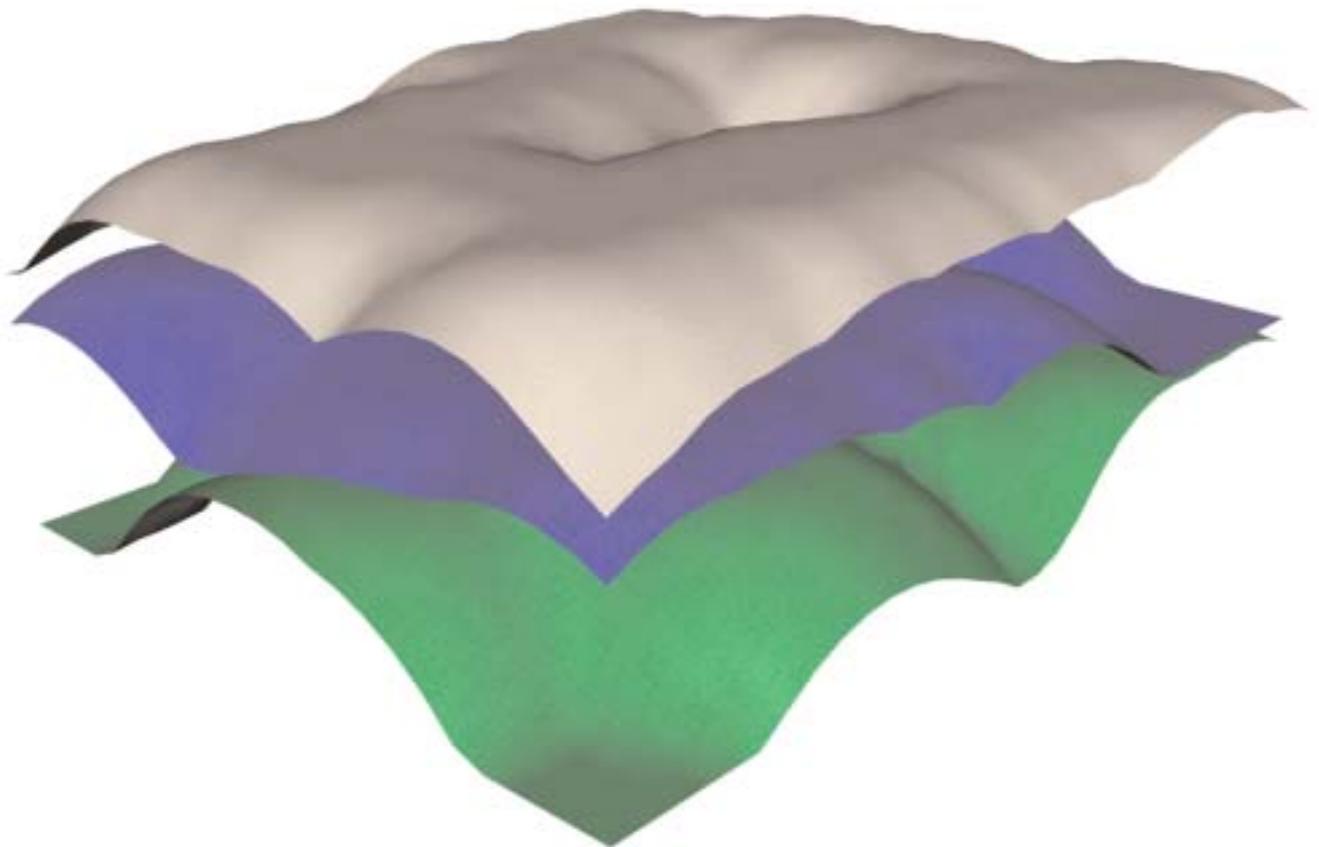


Abb.48: Ebene 03

Durch eine Verfeinerung des Programmes werden die Höhen der oberen Hüllkurven relativ zur darunterliegenden aufgetragen. Die maximale Höhe bleibt bei 4 m. Dadurch entsteht ein direkter Bezug zwischen den Hüllkurven, der in nicht benutzten Bereichen ein Verschmelzen zulässt.

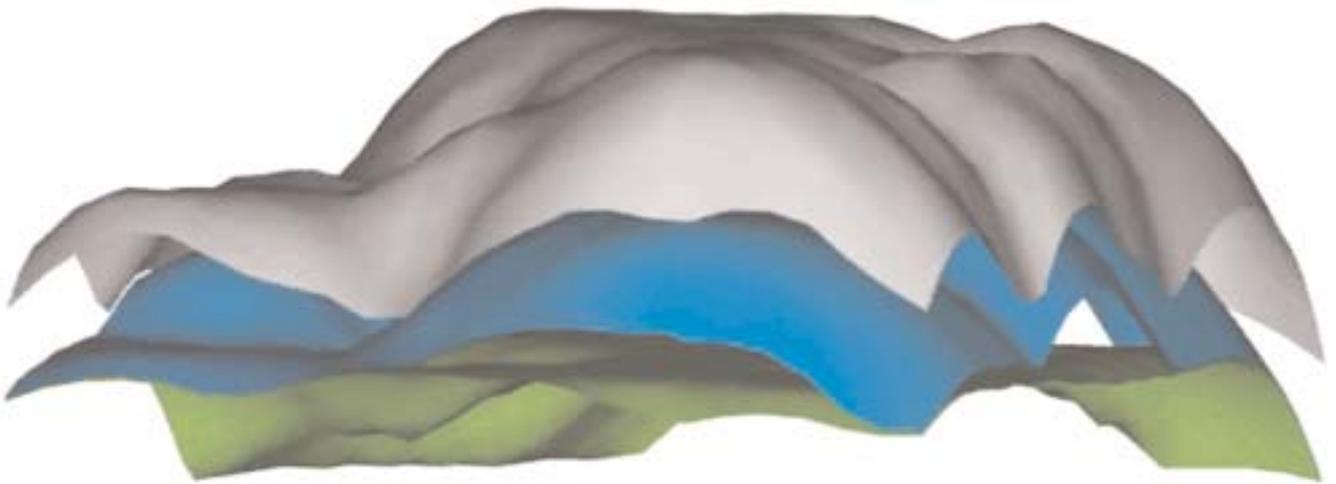


Abb.49: Relativierung des vorhergehenden Beispiels

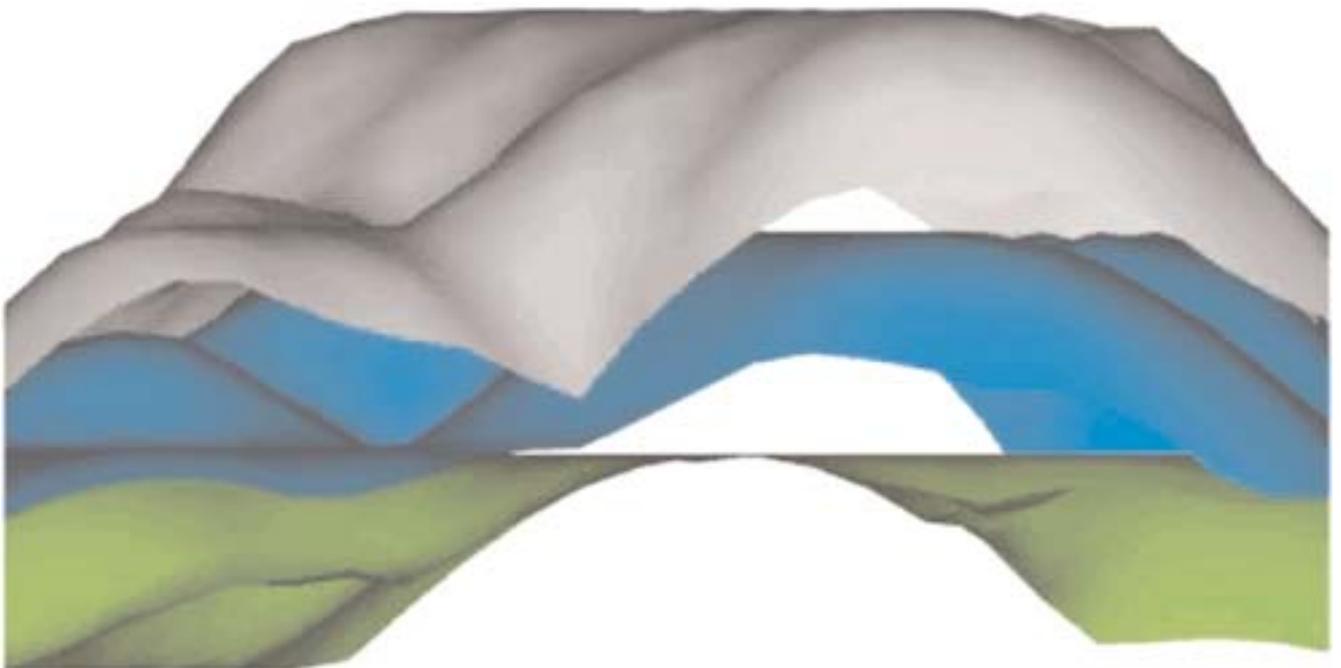


Abb.50: Ansicht der relativierten Stapelung

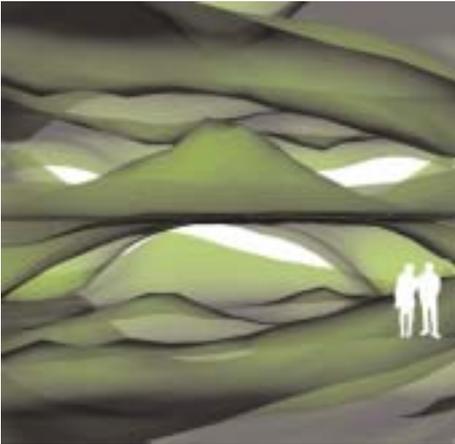


Abb.51: Momentaufnahme aus einer mittigen Simulation.

In der bisherigen Entwicklung wurde immer von der Simulationsebene aus in eine Richtung, nach oben, simuliert. Als zweite Variante gibt es nun die "mittige Simulation", bei der von der fiktiven Ausgangsebene aus sowohl nach oben als auch nach unten eine Raum umhüllende Kurve erzeugt wird. Komplexe Raumgefüge entstehen.

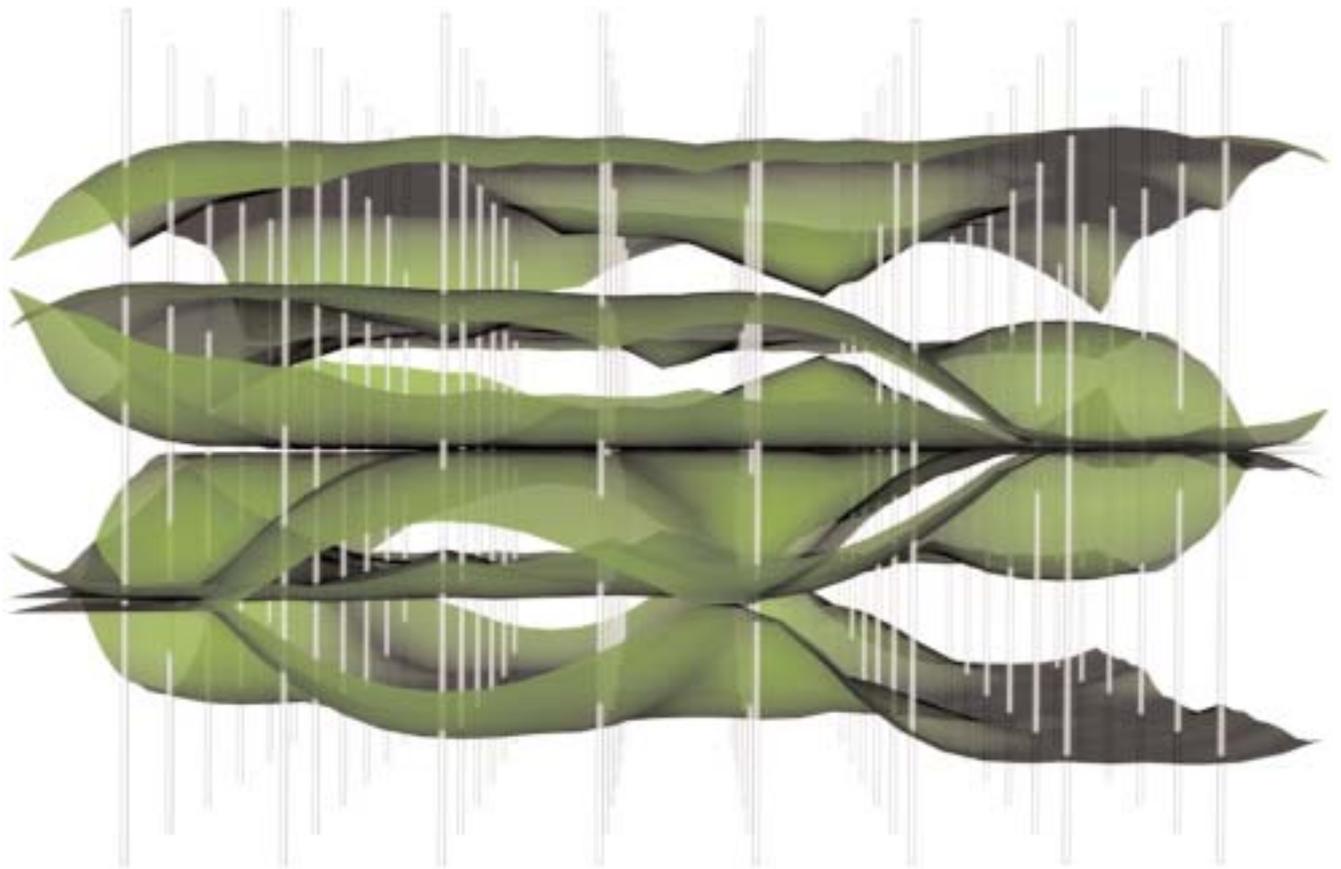


Abb.52.: Stapelung der "mittigen Simulation"

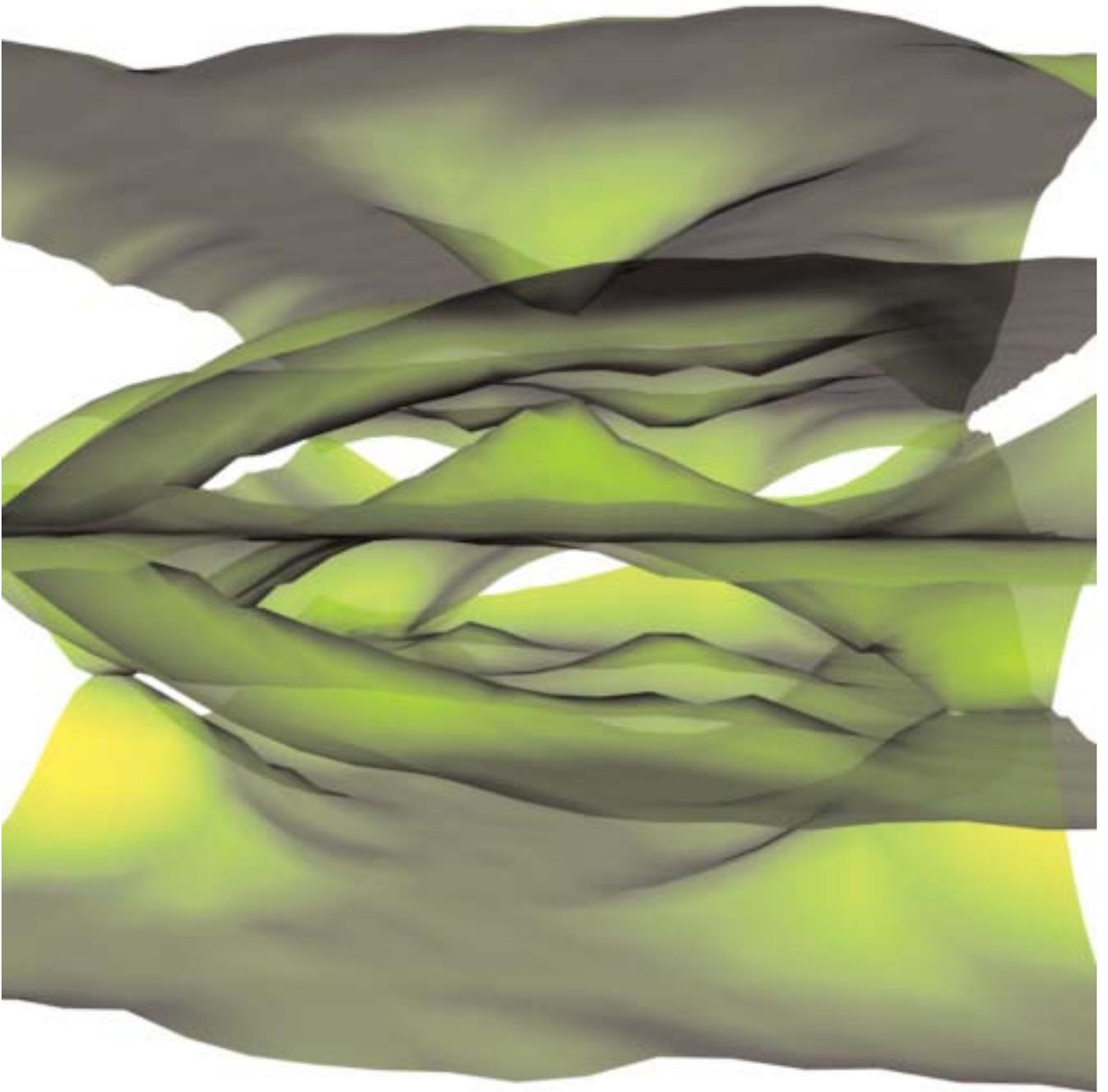


Abb.53: Rendering von den vorhergehenden Ebenen nach mittiger Simulation.



## 04 | Anwendungen

Im folgenden Kapitel werden Anwendungs- und Variationsmöglichkeiten präsentiert. Gegliedert sind die Anwendungen in **small [S]**, **medium [M]**, **large [L]** und **x-large [XL]**. Je nach Maßstab und Dimension der Anwendung gibt es verschiedene Interpretationen der jeweils entstandenen ve.loc.i.ty-Figur.

So sollen die folgenden Beispiele nur einen Überblick der Anwendung und deren Weiterführung verschaffen, keineswegs aber als absolute Ergebnisse gesehen werden. Gleichzeitig werden Probleme und Besonderheiten zu den einzelnen Projekten erörtert.

Abb.54: Anwendungen mit verschiedenen Maßstäben.

das Wolkenhaus

Anwendung für eine einfache Behausung, einen Messestand oder eine Strandbar.

Ausgehend von der gewünschten Maximalfläche von 5,0x5,0 m werden zwei Hüllkurven nach der relativen Methode konstruiert. Die Simulation läuft mit vier Attraktoren und einer max. Raumhöhe von 2,5 m.

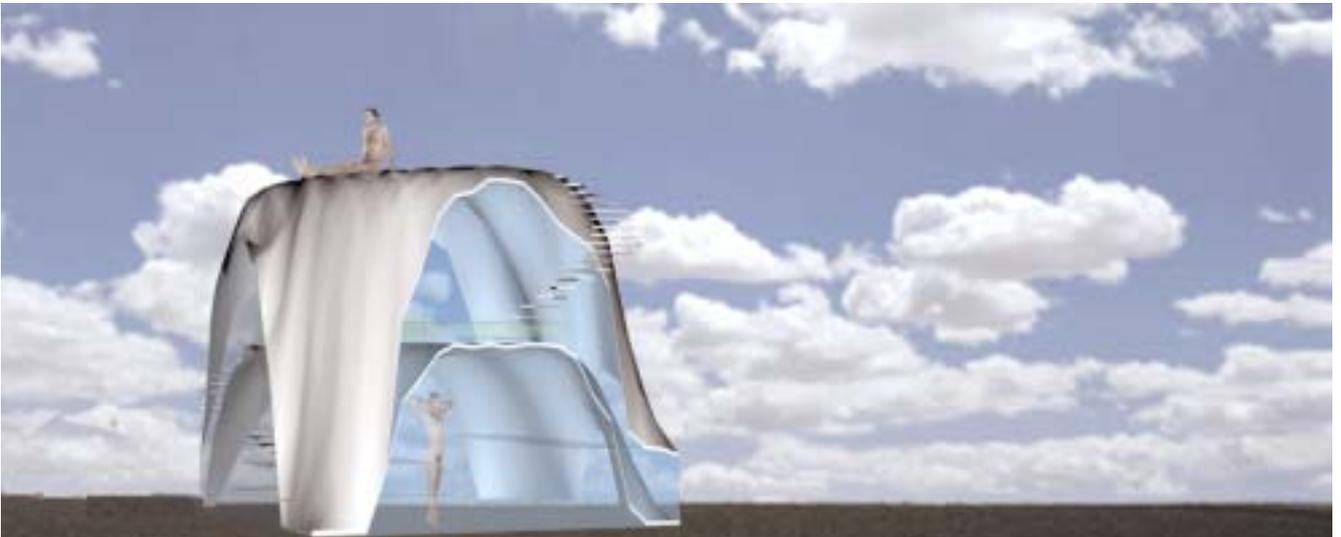


Abb.56: Die Wohnwolke

**OBJEKTDESCHEIBUNG:**

zweigeschoßiges Minimalhaus mit Dachterrasse

Simulationsgröße: 5,0 x 5,0 m

Wohnattraktoren: Bett, Küche, Tisch, Bad

Maße: 25 m<sup>2</sup> /Geschoß

max. Raumhöhe: 2,5 m

Erschließung über zwei Treppen

Konstruktion: Schalenkonstruktion

mit beidseitiger Acrylverkleidung

Konstruktionsstärke: 25 cm



Abb.57: Entstehungsprozess der Wolkenhülle

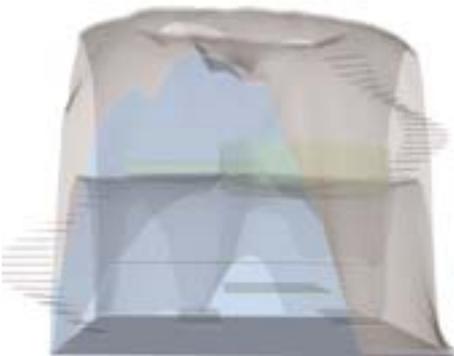
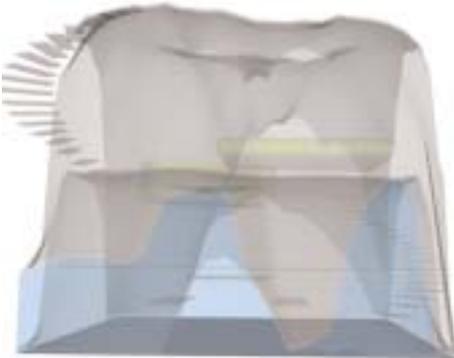


Abb.59: Ansichten der Wohnwolke



Abb.58: Draufsicht

Dieses Beispiel entstand aus einer mittigen Simulation. Die Methode eignet sich nur für eingeschossige Objekte, denn durch die Stapelung mehrerer Räume verliert man den Bezug zur unteren Ebene, welcher die Grundlage für ein mehrgeschoßiges ve.loc.ity-Objekt ist. Durch die ortlose Entwicklung, ausgehend von den maximalen erwünschten Raummaßen (7,0 x 7,0m) gibt es keinen Bezug zu einer Umgebung, was bedeutet, dass es überall, vielleicht auch nur temporär, stehen kann.



Abb.60.: die Wohnkapsel

### OBJEKTDESCHEIBUNG:

Simulationsgröße: 7,0 x 7,0 m

Wohnattraktoren: Bett, Küche, Tisch, Sofa

Maße: ca. 50 m<sup>2</sup>

max. Raumhöhe: 3m

Konstruktion: aufgeständerte Schalenkonstruktion  
mit beidseitiger Acrylverkleidung

Konstruktionsstärke: 25 cm

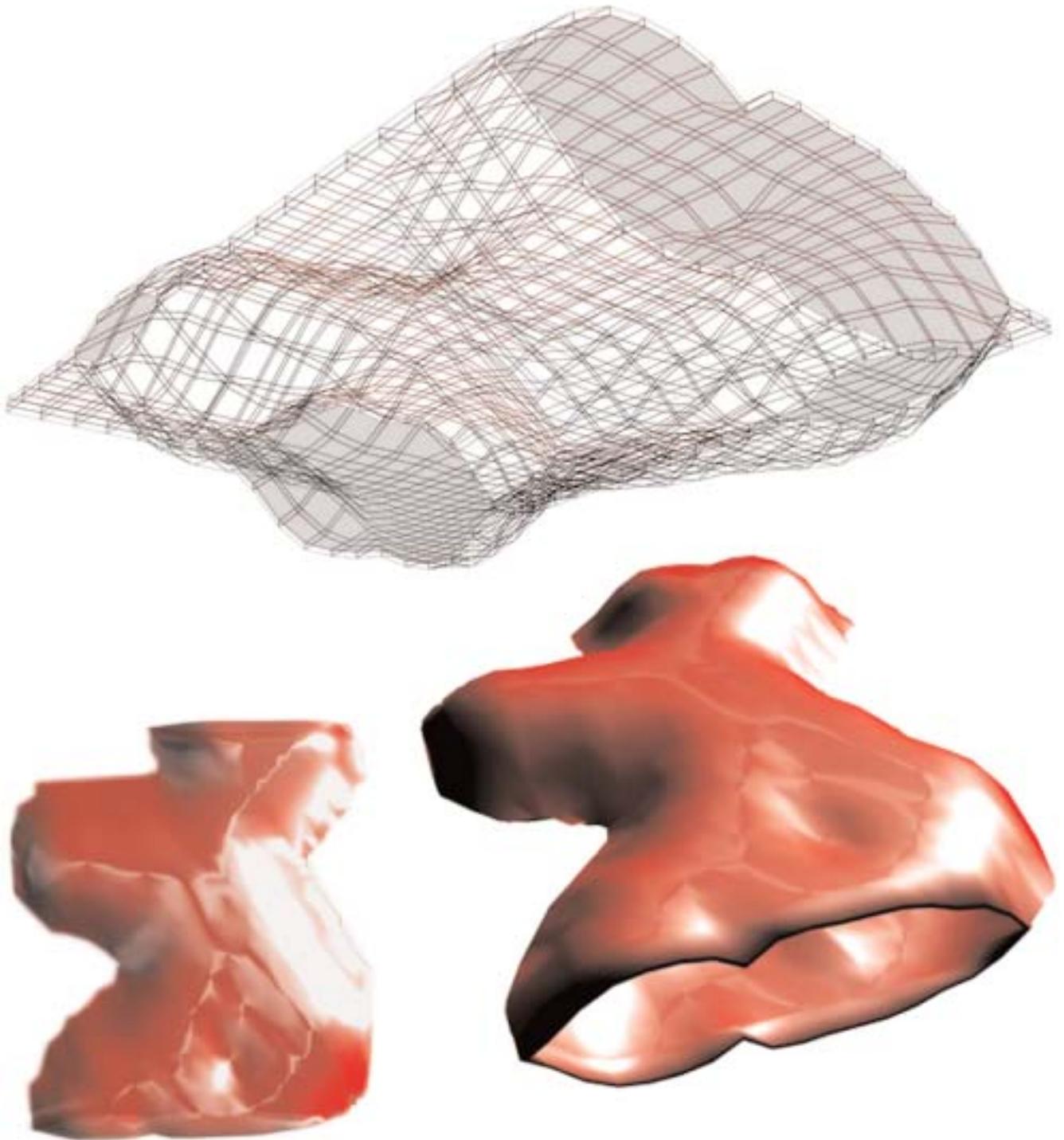


Abb.61: Konstruktion

#### Anmerkung zur Konstruktion

Die Hülle besteht aus einer Schalenkonstruktion. Als Unterkonstruktion dient ein Gittertragwerk aus Stahl. Die beidseitige Verkleidung besteht aus vorgefertigten Acrylteilen mit einer Mindeststärke von 3 cm, welche eine zusätzliche Aussteifung übernehmen.

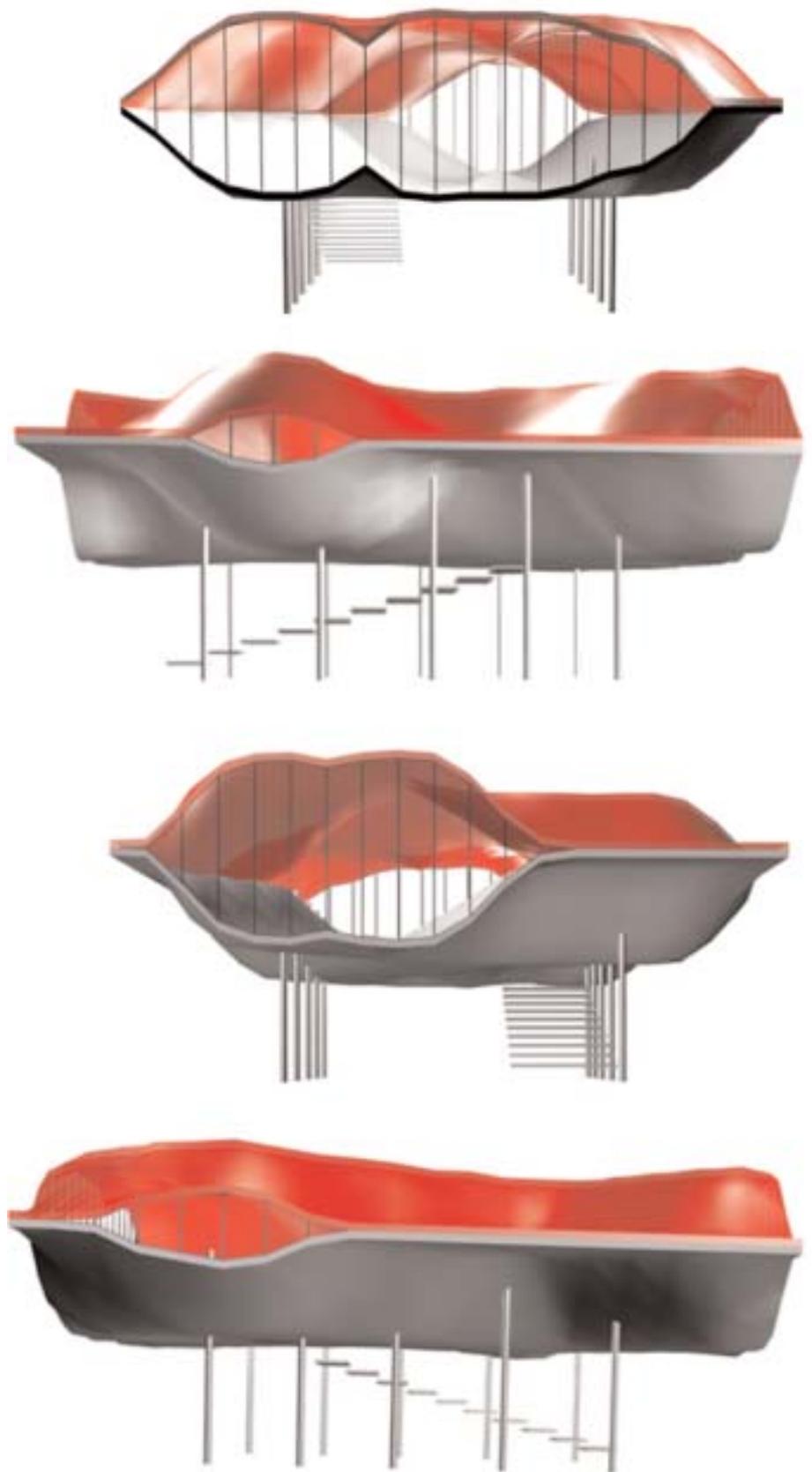


Abb.62: Ansichten

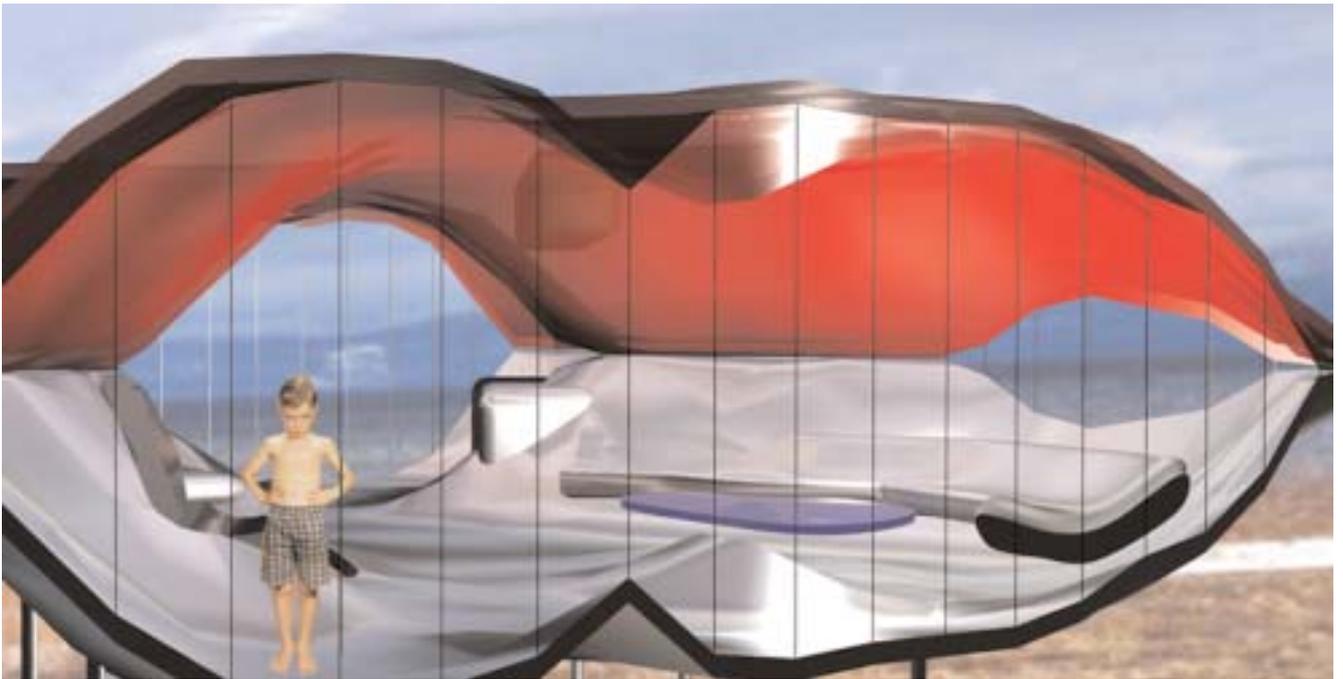


Abb.63: Wohnkapsel



Abb.64: Zugangstreppe mit Eingang in den Kapselraum.



Abb.65: Wohnkapsel mit Frau und Hund



Abb.66: Blickrichtung Bett und Küche

Wie die meisten Anwendungen ist auch die Medium-Anwendung eine ortlose. Der entstandene Raum mit punktuellen Attraktoren ist zudem noch funktionsfrei, so steht dem Benutzer ein Ausstellungsraum, ein akustisch perfekter Konzertraum oder nur ein Leerraum in der Stadt zur Verfügung.



Abb.67: Pavillion am Karlsplatz

**OBJEKTDESCHEIBUNG:**

Simulationsgröße: 25,0 x 15,0 m

Attraktoren: punktuelle Ausstellungsgegenstände

Maße: ca. 375 m<sup>2</sup>

max. Raumhöhe: 4m

Geschoße: 1

Konstruktion: Stütz und Schalenkonstruktion aus Stahl,  
Verkleidung aus Titanblech



Abb.68: Ansichten



Abb.69: Nutzung als Ausstellungsraum



Abb.70: Nutzung als Partyzone oder Konzertraum.



Abb.71: Das Golfpavillion mit Abschußflächen vom Dach

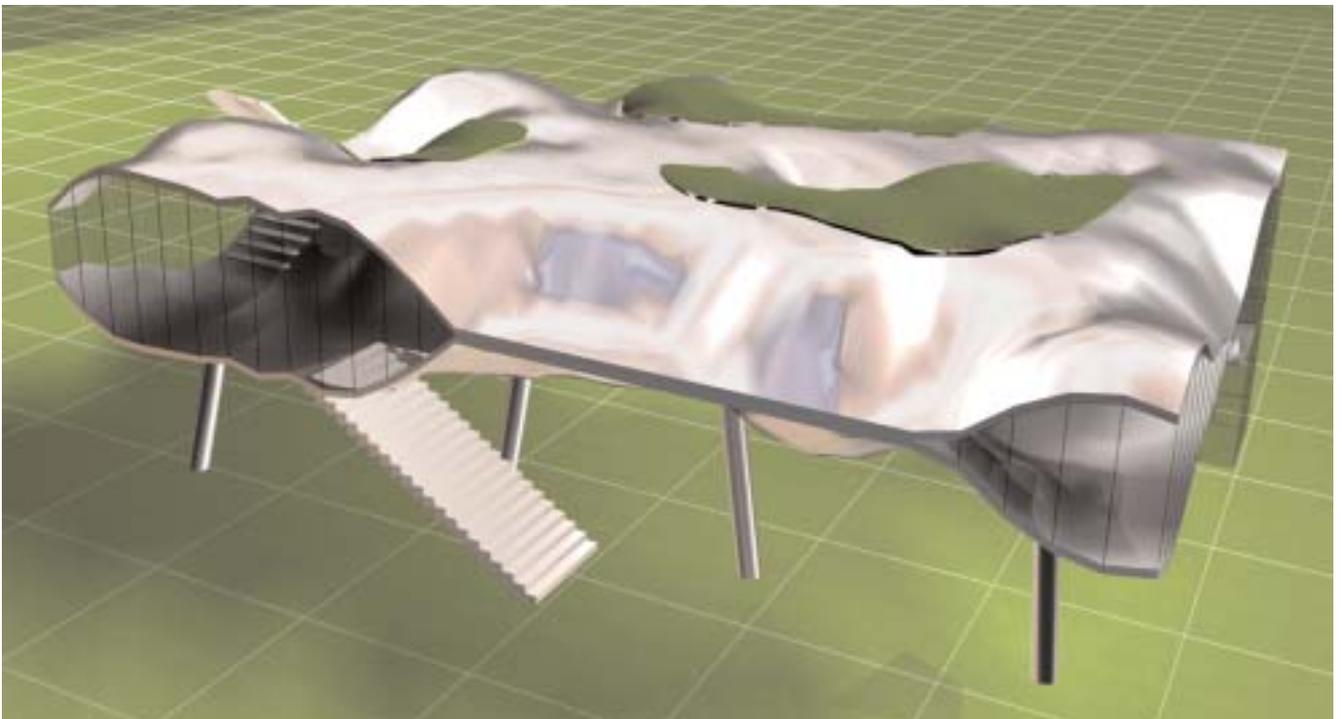


Abb.72: Das Golfpavillion mit Abschußflächen vom Dach

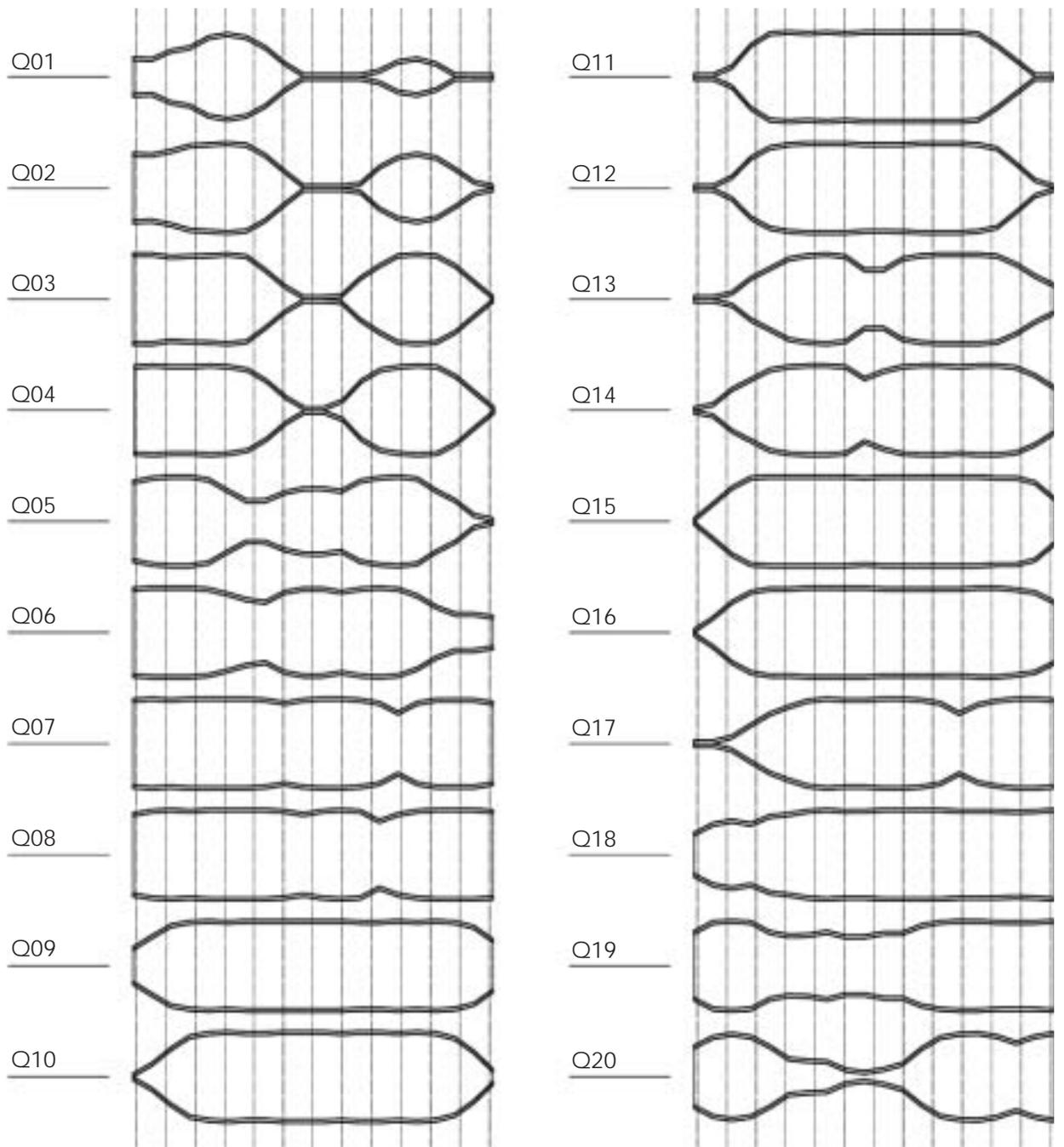


Abb.73: Schemaschnitte - quer

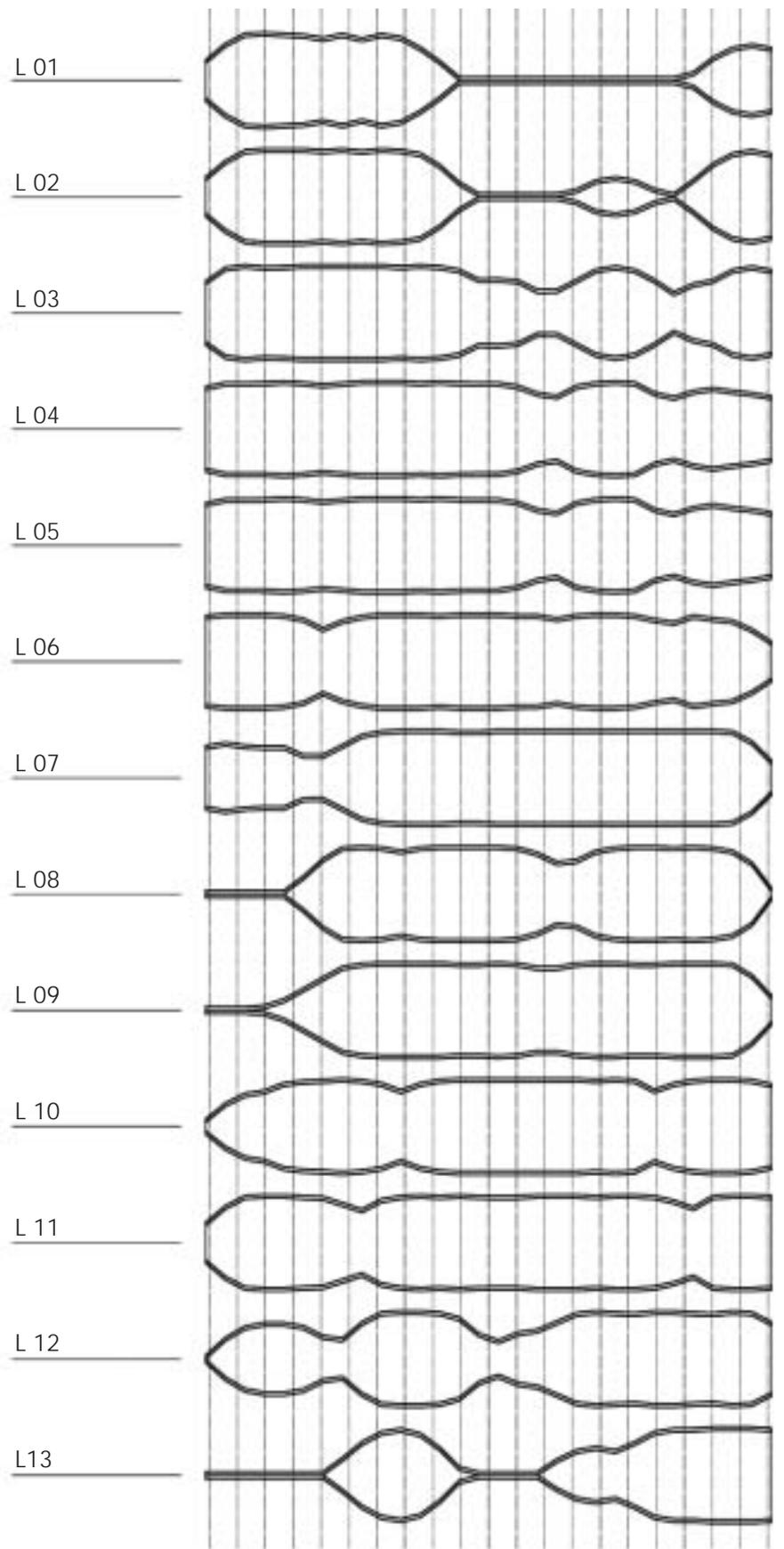


Abb.74: Schemaschnitte - längs

## Modell zum Pavillion

Aufgrund der komplexen Formen wäre eine händische Produktion eines Modelles zu umständlich gewesen, daher wurde folgendes Modell mit einem 3D-Drucker gedruckt.

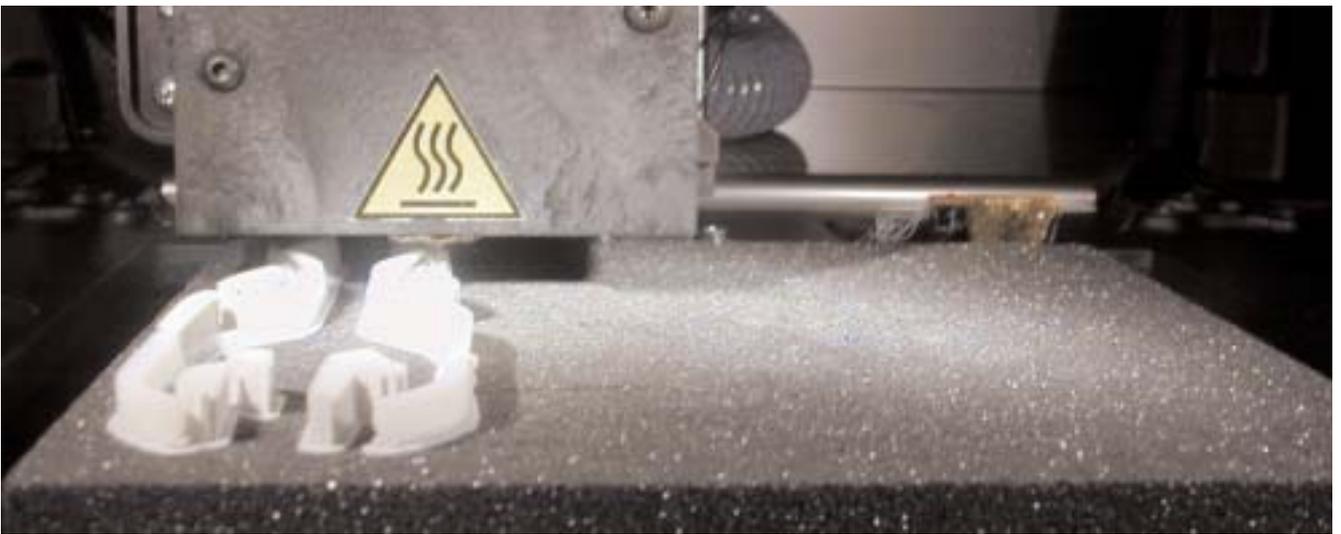


Abb.75: Erste gedruckte Schichten am "Dimension"-Plotter - GBL&E



Abb.76: close up / Innenraum des fertigen Modells

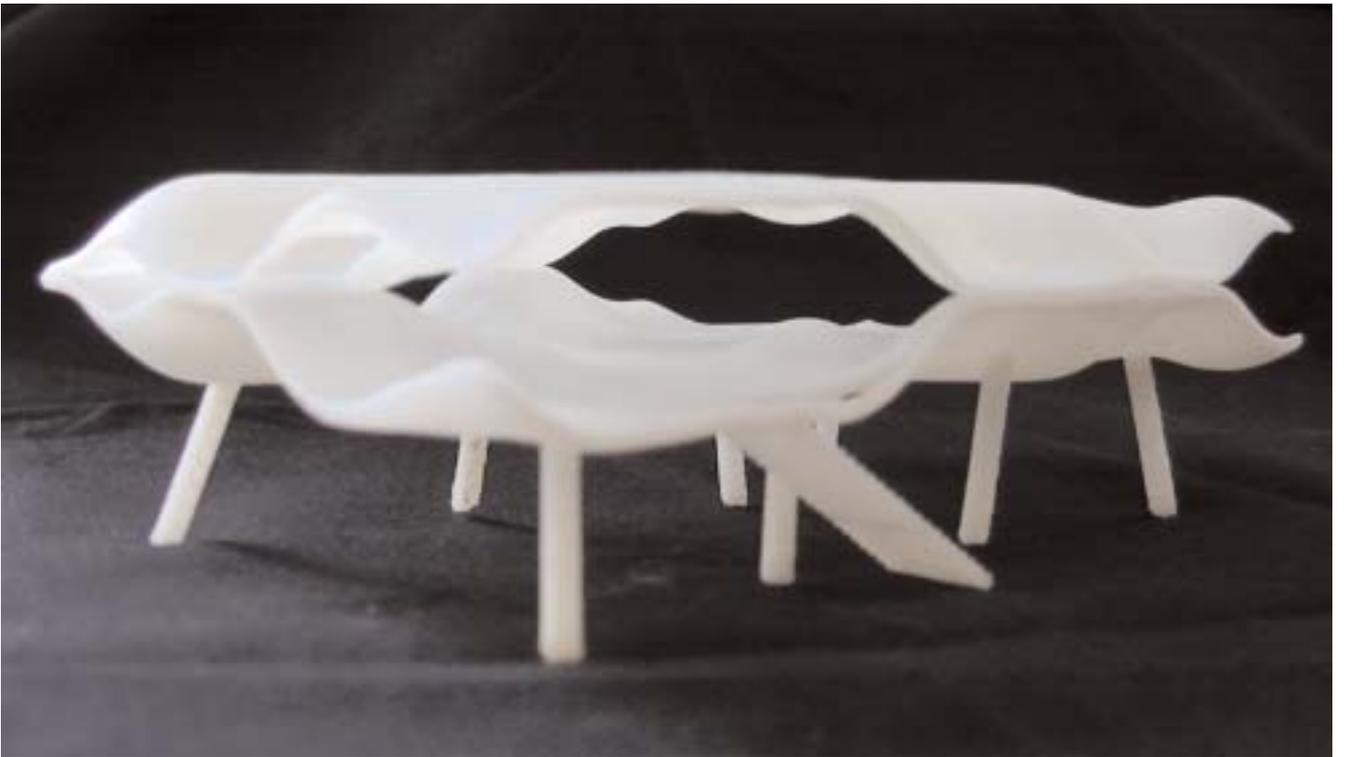


Abb.77: Das fertige Modell



Abb.78: Das fertige Modell

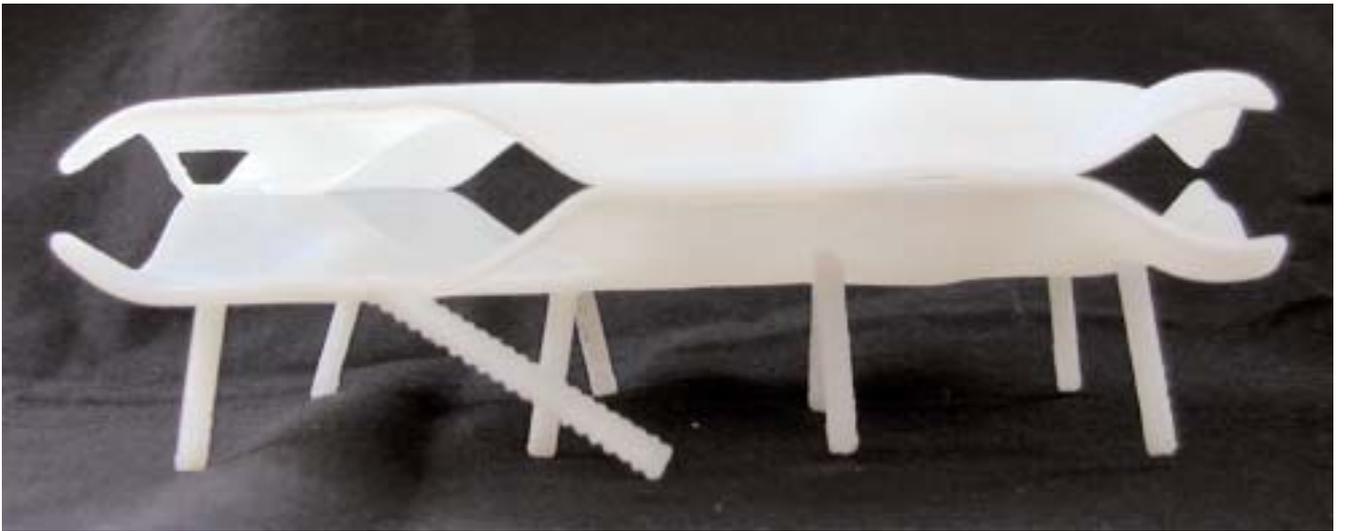


Abb.79: Das fertige Modell

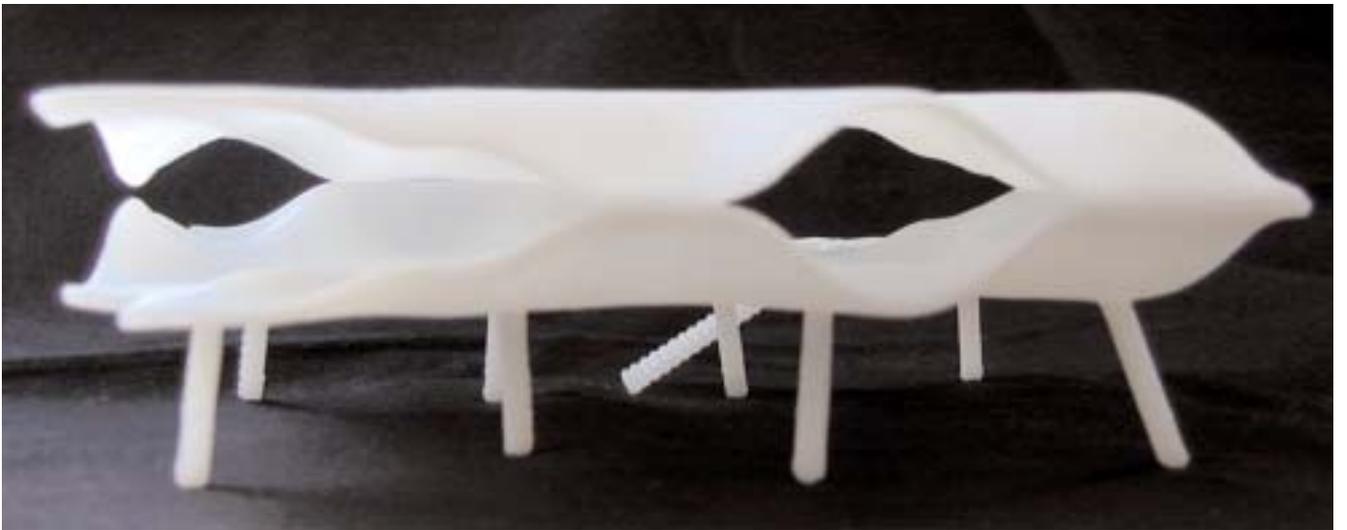


Abb.80: Das fertige Modell

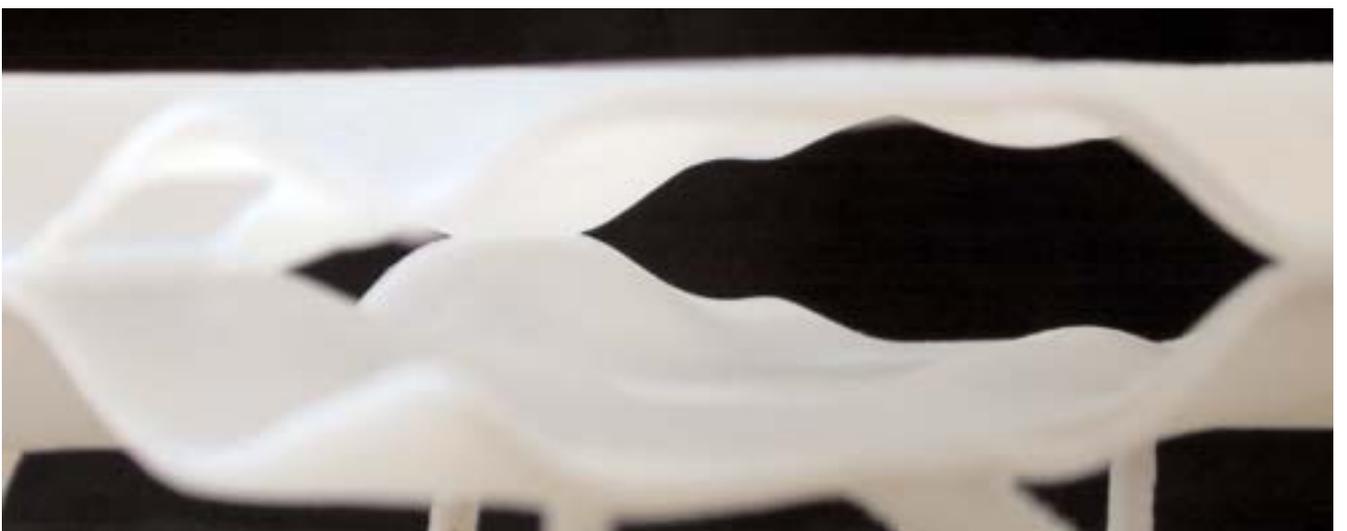


Abb.81: Das fertige Modell

die Baulücke

Die nächsten zwei Beispiele haben Simulationsgrundflächen von 40 x 20m und 20 x 20m. Sie entwickeln sich bis zu 5 Ebenen in die Höhe.

Die Grundparameter für die erste L-Anwendung in der Baulücke sind ein "relativer Aufbau" der Ebenen und eine maximale Steigung von 40% [angenähert ein Steigungsverhältnis von 3:1], wodurch sich die Ebenen verbinden. Als Attraktoren wurden Leinwände gewählt welche in Verbindung mit den ansteigenden Flächen in einer offenen Kinostruktur resultieren.

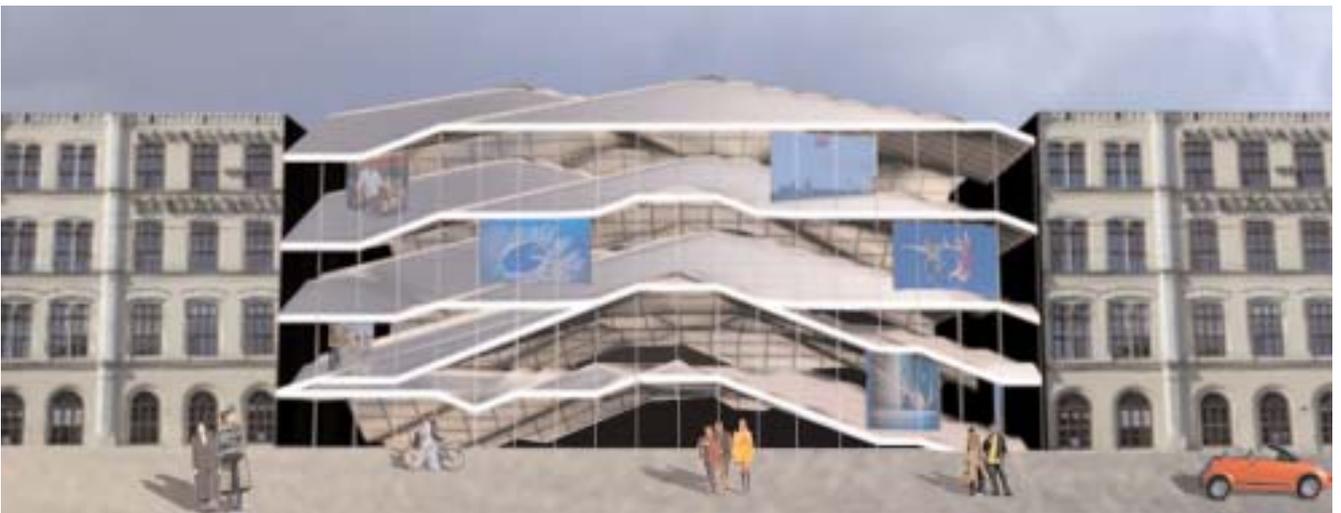


Abb.82: Klno

**OBJEKTBESCHREIBUNG:**

Simulationsgröße: 40,0 x 20,0 m

Attraktoren: Projektionswände

Maße: ca. 800 m<sup>2</sup>

max. Raumhöhe: 4m

Geschoße: 5

Konstruktion: Schalenkonstruktion aus Stahlbeton

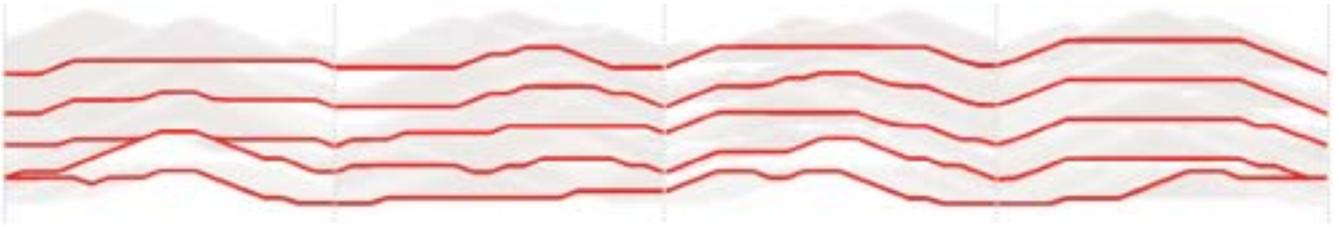


Abb.83:Abwicklung der Ansichten

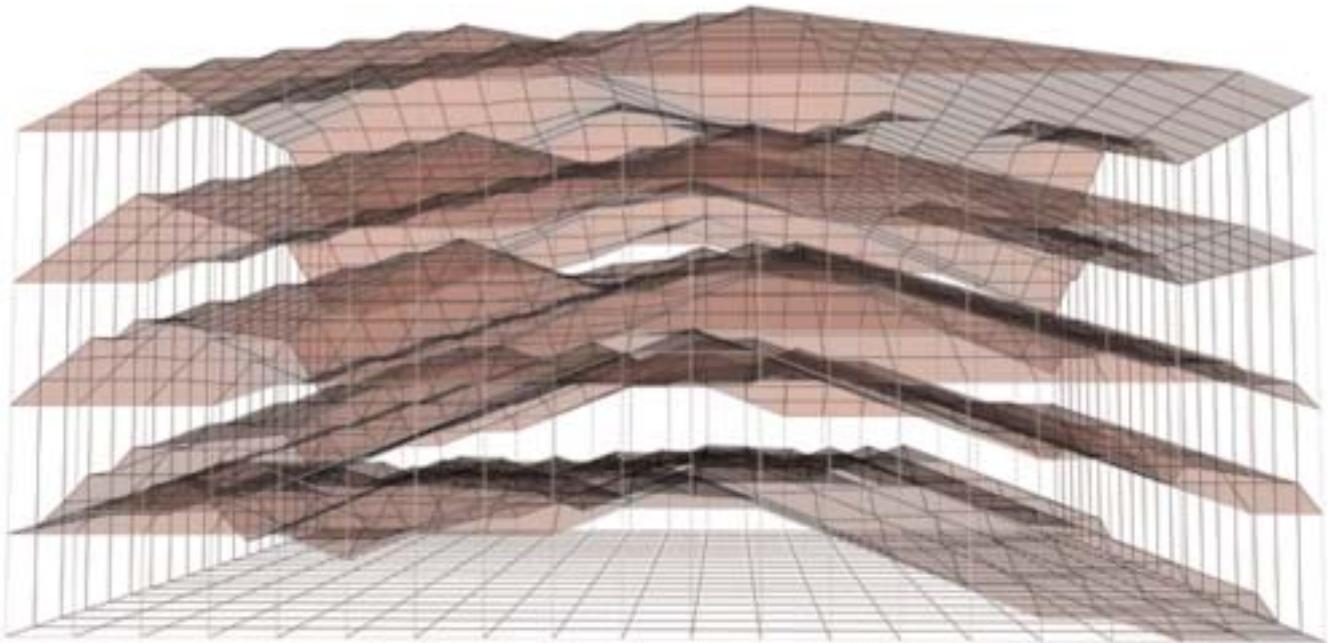


Abb.84: relativer Bezug zwischen den Ebenen



Abb.85: straßenseitige Perspektive

## das Pyramidenproblem

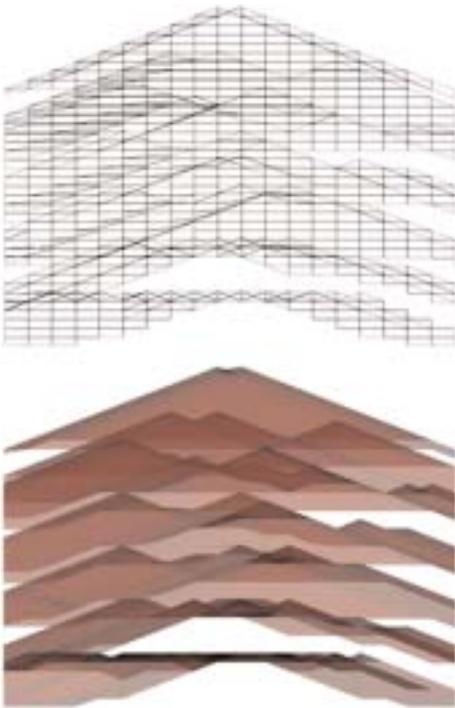


Abb.86: das Pyramidenproblem

Beschränkungen der Steigungsverhältnisse sind absolute Beschränkungen, sie beziehen sich immer auf die aktuelle Simulationsebene. Ansonsten hätte man, bei einer Beschränkung auf 10%, in der zweiten Ebene Steigungen bis zu 20%, in der dritten bis zu 30% usw. Diese absolute Beschränkung bewirkt eine gewisse Loslösung einer Ebene von seinem Vorgänger.

Es kommt zu überhöhten Räumen, und in weiterer Folge zu einer Ausglättung der Ebene. Im Extremfall führt dies soweit, dass nach 7-8 Ebenen nur noch eine Pyramidenform überbleibt.

Daher ist es sinnvoll, diese Kombination maximal über 2-3 Ebenen, bzw. auf größeren Ausgangsflächen anzuwenden. Das Pyramidenproblem kann man auch dadurch umgehen, indem man die Ebenen nicht absolut aufbauen lässt, und zu steil generierte Aufstiege händisch korrigiert.

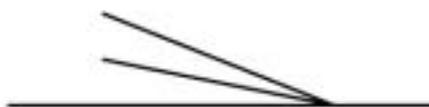


Abb.87: relative Steigungsbeschränkung



Abb.88: Überhöhte Räume und Glättung

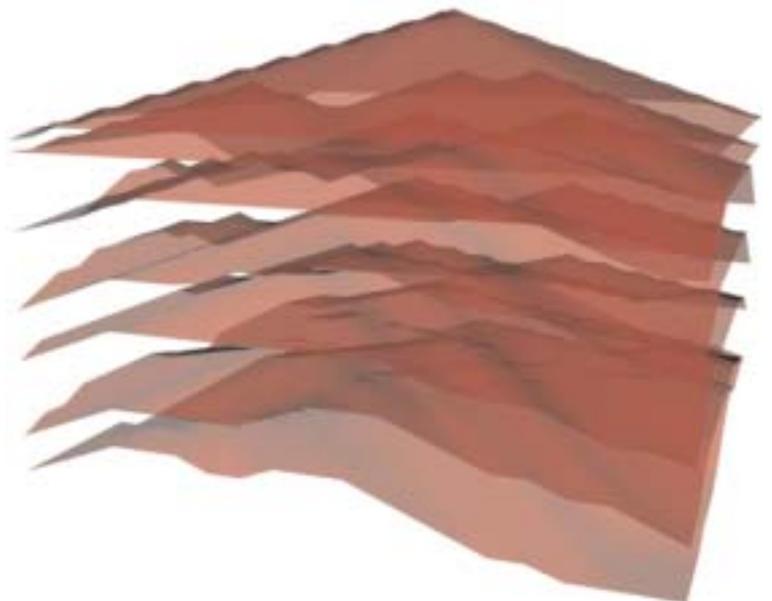


Abb.89: Entwicklung hin zur Pyramide

## Shopping in der Lücke

Die Größe der Baulücke ist 20m x 20m. Simuliert wird mit der absoluten Methode. Als Attraktoren dienen Raumkapseln. Das Resultat ist 5-geschoßig mit einer simulierten Raumhöhe von 4 m.



Abb.90: Shopping in der Lücke

### OBJEKTBESCHREIBUNG:

Simulationsgröße: 20,0 x 20,0 m

Attraktoren: Shopping- und Entertainmentkapseln

Maße: ca. 400 m<sup>2</sup>

max. Raumhöhe: 4m

Geschoße: 5

Konstruktion: Schalenkonstruktion aus Stahlbeton

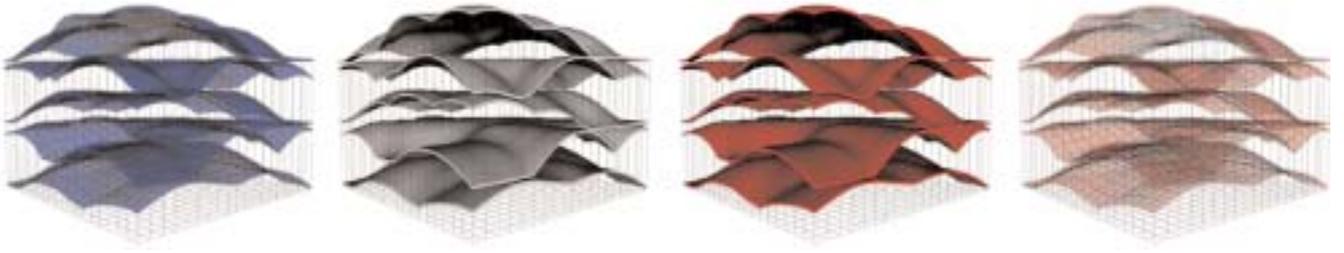


Abb.91: Variation

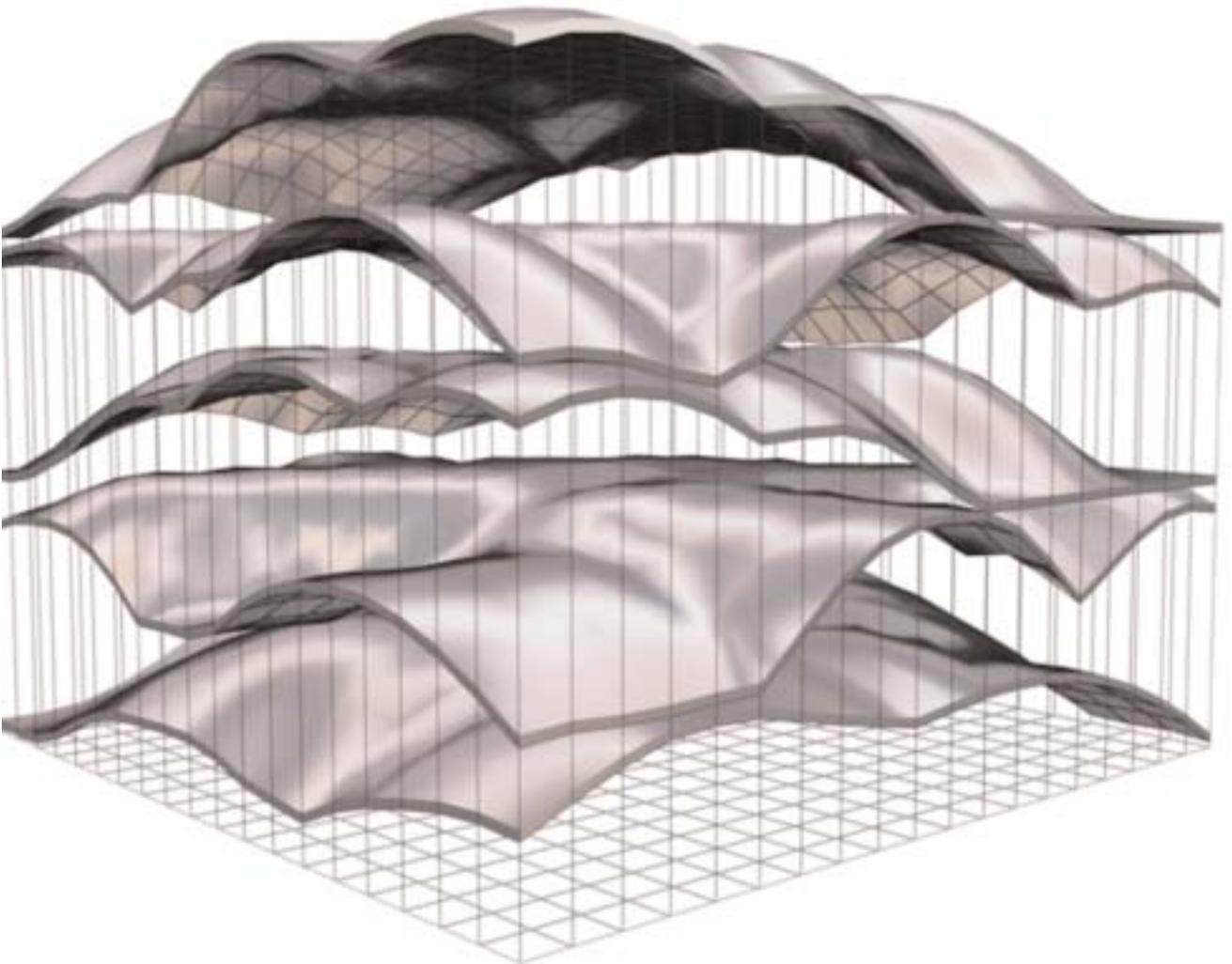


Abb.92: Aufbau

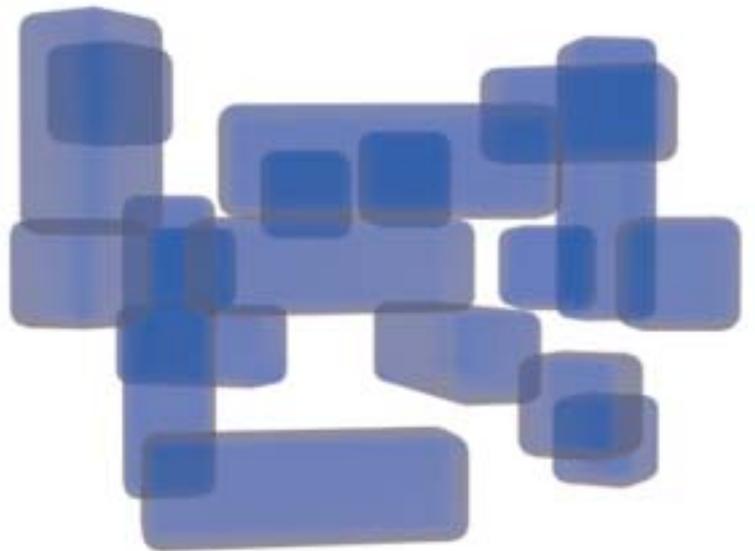


Abb.93: Verbindung der Verkehrsflächen mit den Attraktor-Kapseln.



Abb.94: Blick hinter der Fassade



Abb.95: Innenansicht



Abb.96: Innenansicht

## Anwendung XLARGE [XL]

Eine Anwendung des ve.loc.i.ty-Tools bei Dimensionen von 100 x 100 m aufwärts ist nur sehr reduziert möglich. Zum einen ist für eine so große Simulation bei brauchbarer Rastereinteilung und entsprechend vielen Agentenspuren eine hohe Rechenleistung erforderlich, zum zweiten ist die Übersetzung des Resultates in baubare Architektur sehr schwierig und oft nicht sehr vernünftig.

## Beispiel Europaplatz

Als Beispiel für eine XL-Anwendung wurde ein Ausschnitt des Europaplatzes vor dem Wiener Westbahnhof gewählt. Simuliert wurde in zwei Etappen. Die erste Simulation entstand aus den beiden Gürtel-Fahrbahnen des Gürtels gewählt, wodurch zwei schlauchartige Hügel entstehen. In einem zweiten Schritt wurde die gesamte Fläche vor dem Bahnhof inklusive der Bahnhofshalle behandelt. Dabei wurden die Attraktoren so gesetzt, dass das Ergebnis der Hüllkurve bestimmte Höhenunterschiede überwindet, wie z.B. der zwischen der Halle und den Bahnsteigen.

Das Ergebnis ist nur im Bereich der Bahnhofshalle (Verschmelzung mit der Hüllkurve) und beim U-bahnaufgang der U6 räumlich, über den restlichen Platz ist es eine aus Bewegung resultierende Topographie.

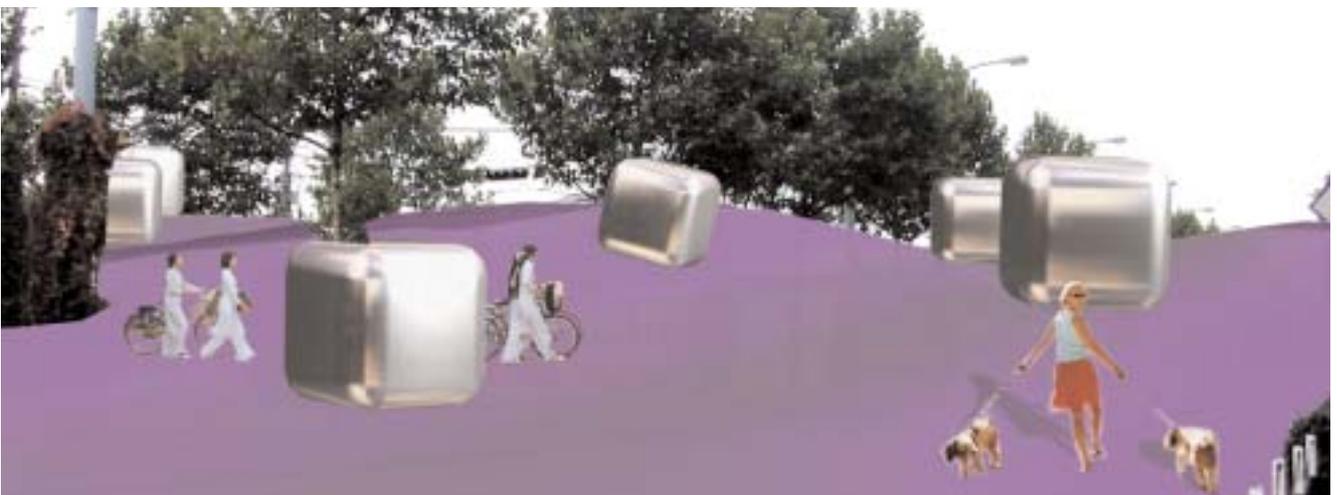


Abb.97: XL-Ebene mit Attraktor-kapseln



Abb.98: Lageplan -Europaplatz

**OBJEKTBESCHREIBUNG:**

Simulationsgröße: 110,0 x 125,0 m

Attraktoren: Raumkapseln von 3 x 3 x 3 m

Maße: 1375 m<sup>2</sup>

Konstruktion: Schüttung mit gewünschten Oberflächenaufbau,  
bei darunterliegenden Räumen Schalenkonstruktionen aus  
Stahlbeton

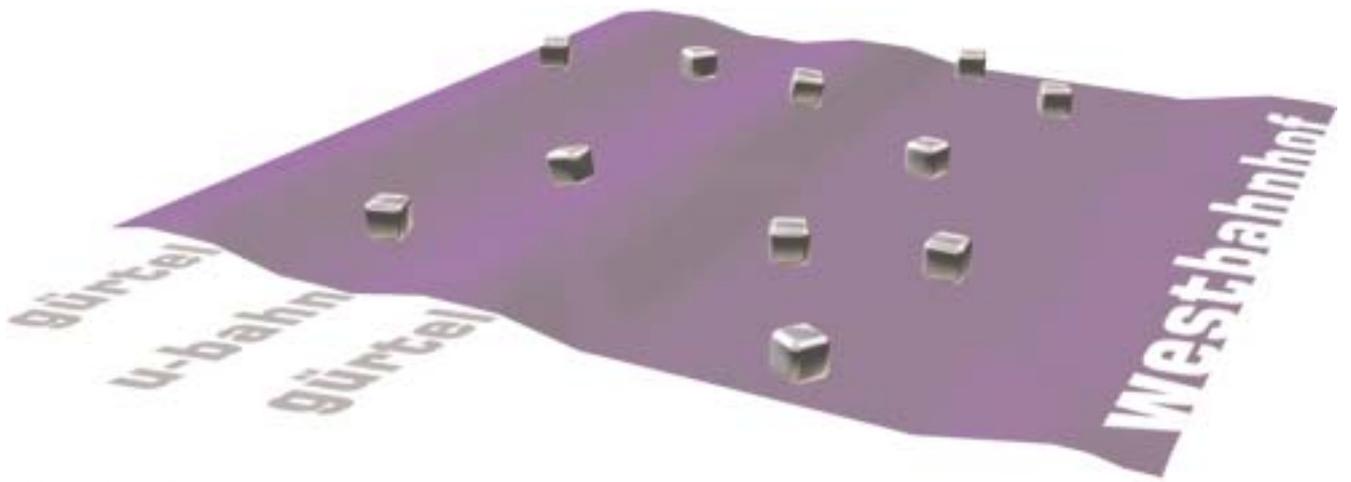


Abb.99: die Hüllkurve



Abb.100: Europaplatz mit Westbahnhof.



**A\* - A-Star**

A\* [ a-star ] | A-Star ist ein gebräuchlicher Pathfinding-Algorithmus, der sehr gerne für Videospiele verwendet wird, er sucht kostengünstigste Verbindung von A nach B unter Berücksichtigung von verschiedenen kostenintensiven Feldern, im konkreten Fall in Raumhöhe und Steigung übersetzt.

**additives System**

Ein additives System ist vergleichbar mit einem aufbauenden Prozessablauf, indem ein Stein über den nächsten gelegt wird. [Küche + Bad + Wohnzimmer + Schlafzimmer = Wohnung] werden mehr und mehr von nichtlinearen Systemen oder Netzwerken abgelöst.

**Agenten**

Agenten sind Platzhalter für den späteren Benutzer. Sie besitzen eine bestimmte Größe [menschliche Intimzone  $r=80\text{cm}$ ,  $h=2.0\text{ m}$ ]. Sie haben die Fähigkeit, sich mittels des A\*-Algorithmus einen Weg von A nach B zu suchen und sich dabei gegenseitig auszuweichen. Die Agenten sind auf verschiedenen Ziele [Attraktoren, Zugänge, Ausgänge] programmiert.

**Attraktoren**

Attraktoren ziehen je nach angebotener Attraktion über einen gewissen Zeitraum verschieden viele Agenten an. Der bespielte Raum eines Attraktors ist abhängig von seiner zeitlichen Nutzung, so expandiert die Fläche um einen Funktionskern je nach Nutzungsintensität. Man unterscheidet Minimalattraktoren (Automaten, Schließfächer, Ausstellungsobjekte u.ä.) bei denen ein Teil der Verkehrsfläche beansprucht wird und den klassischen Attraktor, bei dem ein vordefinierter Raum bespielt wird (Cafe, Geschäfte oder Kino)  
sh. auch: Funktionskern, Agenten

## Designtime

Designtime, im architektonischem Sinne ist jene Phase, in der man die Rahmenbedingungen (Bauplatz, Grundriss, gewünschte Dichte, Anzahl der Attraktoren, ...) und die Parameter eines Durchlaufes festlegt. Man gestaltet das Projekt. Treten hierbei Fehler (design-time errors) auf, lässt sich der Schritt in die runtime nicht ausführen.

## Dichte Lücke

"dichte lücke" ist die Projektbezeichnung für eine konzentrierte Raumnutzung von Verkehrsfläche und Attraktoren (plugins), bei dem die Möglichkeit des schnellsten Weges für jeden einzelnen Benutzer aufrecht bleibt. (siehe: gleichnamiges Projekt Grasl-Falkner 2001 - Institut für Hochbau)

## F.O.L.E.D. - Stoffe

F.O.L.E.D.-Stoffe (Flexible Organic Light Emitting Display) sind flexible Aktivmatrixdisplays, welche faltbar sind. Entwickelt an der Princeton-Universität. Nach weiterer Entwicklung als Projektionsfläche bei der Moverfassade verwendbar.

sh.auch: mover

## Funktionskern

Der Funktionskern ist das Herz eines Attraktors, genauer gesagt der Leistungsbringer (Badewanne oder Dusche im Badezimmer). Bei Minimalattraktoren ist der Funktionskern der Attraktor. (z.B. Automaten)

## Hüllkurve

Die Hüllkurve ist die von ve.loc.i.ty generierte Figur, welche vom 3d Studio als Mesh (Netz) übernommen wird. Diese Figur ist wertfrei zu verstehen und steht für Interpretation offen.

## Laufzeitfehler | run-time-error

Ein Laufzeitfehler (run-time-error) führt zu einer Unterbrechung der "runtime". Festgestellt wird er während der Benutzung. (z.B. zu steile Treppen oder Türen ohne Anschlussraum). Ein Laufzeitfehler wird durch eine ungültige Variablenzuweisung (z.B. hochbautechnisch) hervorgerufen, die in der design-time nicht erkannt wurde. Durch den run-time-error kommt es zu einer Nullzeit. Das Programm muss in der "design-time" überarbeitet werden.

**level of service**

level of service ist der Grenzbereich, an dem die Agentendichte und die damit zusammenhängende Attraktorenzahl so hoch ist, und keine komfortable Nutzung der Verkehrsflächen mehr gewährleistet ist. Der Begriff kommt eigentlich aus dem Straßenbau und wird an den jährlichen Stunden der Verkehrsüberlastung gemessen, woraus man schließlich den Straßenquerschnitt bemessen kann. Kommt es im vorliegenden Fall zu einer Überlastung, so wird die Attraktorenfläche reduziert.

**mittige Simulation**

Die mittige Simulation ist eine Simulationsmethode des ve.loc.i.ty-Tools, bei der die Hüllkurve sowohl nach unten als auch nach oben generiert wird. Diese Methode ist nur für eingeschobene Objekte geeignet, sh.auch Simulationsebene, ve.loc.i.ty

**mover**

Mover - ein Programm, welches eine räumliche Wahrnehmung von Bewegung ermöglicht, und durch die Kombination vom statischem Medium (Lein-) Wand und einem laufenden Film eine vielfältigere Sehweise entstehen lässt.

**Oberflächenerweiterung**

Unter Oberflächenerweiterung versteht man die optimierte Nutzung des gebauten Raumes. Im vorliegenden Fall entsteht diese Verdichtung durch das spezielle Kombinieren von Verkehrsfläche und Attraktoren, wodurch verkehrslastige Flächen wie Gänge und Treppen reduziert werden.

**offenes Raumgefüge**

In einem offenen Raumgefüge gibt es keine klaren Trennungslinien zwischen verschiedenen Zimmern. Eine freie Funktionszuordnung in einem flächigen Erschließungssystem wird möglich.

**Ort**

Prinzipiell ist das System nicht an einen Ort gebunden. Durch das Implantieren in bebaute Strukturen [Baulücken, urbaner Platz ...] ist die Entwicklung von bestimmten Faktoren wie z.B. der Umgebung oder der Frequenz von Agenten an diesem Ort abhängig. Diese Faktoren wirken sich schließlich in der Simulation aus.

### Prozessing time

Die "prozessing-time" teilt sich auf in run-time (fertig bezogenes Objekt) und design-time (Simulations- und Planungsphase).

### Pseudocode

Ein Pseudocode ist die Struktur eines Programms. So geschrieben, dass sie leicht in eine echte Programmiersprache umgesetzt werden kann. Pseudocode kann nicht kompiliert oder ausgeführt werden und es gibt keine Format- oder Syntax-Regeln. Man kann Pseudocode schreiben, ohne zu wissen, in welche Programmiersprache das Programm letzten Endes umgesetzt wird.

### Pyramidenproblem

Das Pyramidenproblem ist ein Phänomen, welches bei einer ve.loc.i.ty-Simulation mit absoluter Beschränkung der Steigungsverhältnisse auftreten kann. Diese Art der Simulation hat die Tendenz, nach jedem Durchlauf einer Pyramidenform ähnlicher zu werden.

### relativ aufbauend

"Relativ aufbauend" ist eine Methode des ve.loc.i.ty-Tolls bei gestapelten Ebenen. Dabei bezieht sich jede Ebene auf die darunterliegenden und es kommt in bewegungsarmen Bereichen zu Verschmelzungen. Im Gegensatz dazu gibt es die absolute Methode des Stapelns, bei der die Hüllkurven innerhalb der maximalen Simulationshöhe generiert werden.

### Raumfilm

Der Raumfilm wird durch den Mover laufend neu entwickelt und führt zu einer sich ändernden Wahrnehmung des Raumes. Im übertragenen Sinn auf die Architektur wird Gebautes zur "Leinwand" und der Benutzer gleichzeitig zu Akteur und Regisseur.

sh.auch: mover

### rekursiver Prozess

Ein sich selbst aufrufender Prozess. Wird bei ve.loc.i.ty angewendet, um eine Situation solange zu überarbeiten, bis eine Abbruchbedingung erfüllt wird.

## Runtime

Runtime im architektonischem Sinn ist die Phase der Gebäudenutzung. Treten in dieser Phase Fehler (runtime errors) auf, lassen sie sich nur unter hohem Aufwand beheben. Deshalb wird hier eine Simulation zwischengeschaltet, um Fehler sobald als möglich zu erkennen und mit geringem Aufwand beheben zu können.

## stabiler Zustand

Der stabile Zustand ist der Punkt an dem die Simulation baubar ist. Durch das ständige Wechselspiel zwischen Agent und Attraktor kommt es immer wieder zu Laufzeitfehlern, welche in der designtime korrigiert werden. Die simulierte Bewegung des Gebäudes ist zeitlos und soll nicht als direkte Übersetzung verstanden werden, vielmehr erreicht man über das Überlagern der Überlagerungen immer idealere Bilder, bis ein Grenzbild erreicht wird, welches dem stabilen Zustand am ehesten entspricht.

## Simulationsebene

Die Simulationsebene ist die gewünschte "Baufläche" die durch bauliche oder fiktive Grenzen (z.B. Baufluchtlinien) vordefiniert ist. Zur eigentlichen Simulation in ve.loc.i.ty bedarf es einer Aufarbeitung dieser Simulationsebene zu einer Autocad .dwg, der eine bestimmte Layerstruktur zugrunde liegt.

## Steuerungsverhalten

Steuerungsverhalten (steering-behaviours) ist das programmierte Verhalten, das den Agenten die Möglichkeit gibt, sich fortzubewegen. Implementiert wurde ein Verhalten zum Ansteuern eines vorgegeben Punktes, sowie eines, welches Kollisionen vermeidet.

## ve.loc.i.ty

ve.loc.i.ty dient der Simulation von Agentenströmen zwischen Zugängen und Attraktoren. Das Programm baut auf einer Autocad-zeichnung auf und wird mit bestimmten Parametern, wie Lage der Zugänge und Attraktoren oder der Anzahl der Agenten gespeist. Unter Einbindung der mover-Funktion werden Daten über die Bewegungsströme an ein max-script weitergeleitet, welches ein mesh (Netz) im 3d-Studio generiert. Neben den statischen Resultaten die die Informationen der Agentenbewegung widerspiegeln, werden auch Filme der Simulation und des entstehenden Meshs mitgeschnitten.

**Zeit**

Parameter der einer bestimmten Handlung zugeschrieben ist.

**zeitlos**

zeitlos | durch die ständige Manipulation und Entwicklung der Attraktoren im offenem Raumgefüge ist kein Zeitpunkt eines "fertigen" Resultats festzulegen, vielmehr haben wir es mit Momentaufnahmen zu tun, anhand derer man die Veränderungen während der designtime als auch in der runtime ablesen kann. Zeitlos auch im Sinne von Epochen- und stilfrei.

sh. auch: designtime, runtime, Laufzeitfehler

## Verzeichnis der Abbildungen

Abb. 00 Portraits von Brigitte Gaggi 2001

Abb. xx Freecard Institut für Gebäudelehre SS02

Abb. 30 Old Campton Street, London | Hulton - Archive

Abb. 32 Beverly Center Shoppingmall, Los Angeles, USA  
Photo by Don Smetzer

Abb. 34 ISS inside | Internetquelle: <http://www.nasa.gov>