

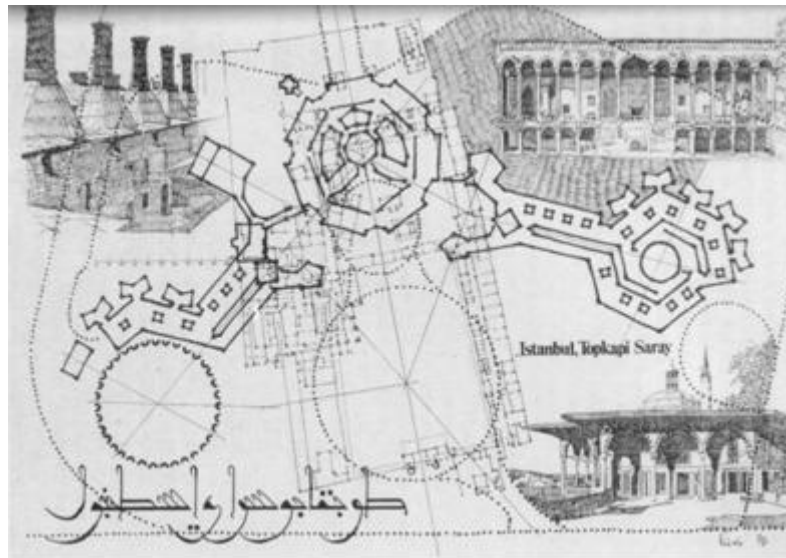
1 Das „Game Design Dokument“ als Gattungsbegriff

Auf der deutschen Spieleentwicklerkonferenz Quo Vadis 2006 in Oberhausen sollte in einer Roundtable-Veranstaltung „Der tiefere Sinn (und Unsinn) von Game Design Dokumenten“ diskutiert werden. Nach meinem Dafürhalten ist diese nach knapp zwei Stunden ohne konkretes Ergebnis oder Resümee zu Ende gegangen.

Es wurde diskutiert, ob das Game Design Dokument für den Designer, für das Team oder für den Publisher entsteht; ob es besser in einem Wiki oder in Word verfasst wird und ob es nach der Fertigstellung ‚starr‘ oder ‚lebendig‘ sein sollte.

In diesem Aufsatz möchte ich mein persönliches Resümee ziehen: Alle genannten Funktionen und Formen stehen nicht gegeneinander, sondern zeigen die Vielfalt der möglichen Aufgaben für Game Design Dokumente - versteht man darunter umfassend die inhaltliche Dokumentation des Game Designs und nicht einen spezifischen Teilausschnitt.

Das „Game Design Dokument“ (GDD) wäre damit ein Gattungsbegriff, so wie die „Architekturzeichnung“ alle zeichnerische Dokumentation zu einem Gebäude umfasst und nicht nur eine spezifische Zeichnung.



Repräsentationsblatt zum Regierungsgebäude, Bernd Riede 1976

Anhand eines Vergleiches mit der Architekturzeichnung möchte ich all die auf dem Roundtable genannten Teilaufgaben dieser Dokumentation innerhalb des Projektverlaufs zeitlich und funktionell darstellen und auch auf die besonders hitzig diskutierten formalen Aspekte eingehen.

Dem voran stelle ich jedoch zwei wichtige Aussagen aus der Roundtable-Diskussion, die nicht deutlich genug hervorgehoben werden können:

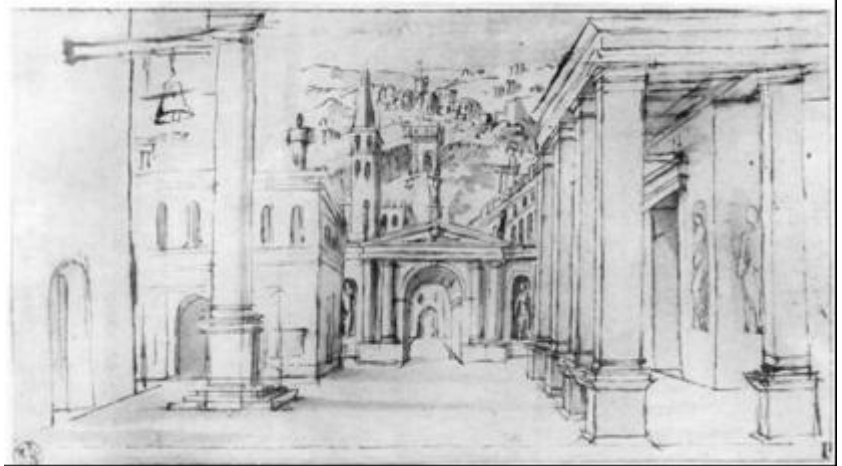
- Game Design Dokumente sollten kein Selbstzweck sein, sondern sich immer der Lösung ihres (jeweils) spezifischen Problems unterordnen. Dazu gehört eine gute Kenntnis des Adressaten und des Umfelds aus Projekt- und Firmenprozessen.
- Kein Game Design Dokument ersetzt die direkte Kommunikation, selbst nicht in den Fällen, in denen es nach Meinung des Autors die gestellte Frage abschließend beantworten würde.

1.1 Das Designer-Notebook: „Code des Designers“

Eine absolut unverzichtbare Aufgabe für GDD ist es, dem Designer selbst als Erinnerungsquelle und ersten Prüfstein eines Designs zu dienen.

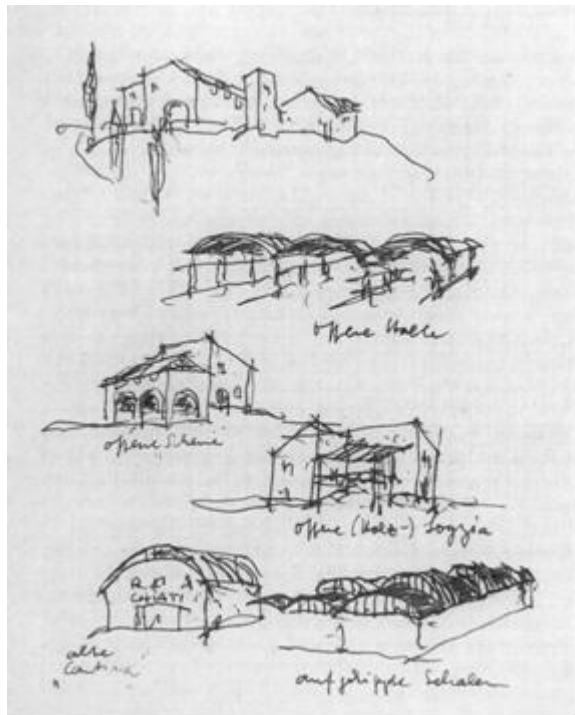
Manche Ideen, die als reine Gedanken im Kopf des Designers noch prima funktioniert haben, verweigern in schriftlicher Darstellung plötzlich den Dienst. Die schriftliche Darstellung erleichtert es, andere Ideen (Ziele, Einschränkungen, etc.) prüfend daneben zu halten und die Idee im Gesamtkontext des Projekts zu betrachten.

Mag dies am Anfang noch im Kopf funktionieren, wird dies mit zunehmender Komplexität eines Designs immer schwerer - ich will behaupten: Selbst bei kleinen Projekten ist es unmöglich, alle Features und ihre Abhängigkeiten allein im Kopf zu verwalten. Es ist daher notwendig, diese schriftlich, tabellarisch, bildlich oder per Diagramm zu dokumentieren, auch um sie später zu erinnern.



Entwurfsskizze. Bernardo Buontalenti im 16.Jh.

Blicken wir über den Tellerrand in Richtung Architektur, dann finden wir dort als vergleichbares ‚Tool‘ den Skizzenblock mit zeichnerischen (und schriftlichen) Notizen.



Skizzenblatt, Otto Steidle 1979

Minimale Anforderung ist es, dass der Verfasser selbst sie lesen kann und sich auch nach längerem zeitlichem Abstand in der Struktur seiner Aufzeichnungen noch zu Recht findet. Da es sich um ein persönliches Werkzeug handelt, werden Inhalt, Umfang und Form zunächst durch den einzelnen Designer geprägt.

Spätestens in größeren Teams oder Firmen wird darüber hinaus die Anforderung bestehen, dass andere Game Designer diese Notizen ebenfalls lesen können sollen. Dies ist der minimalen Anforderung in jedem Falle zuträglich.

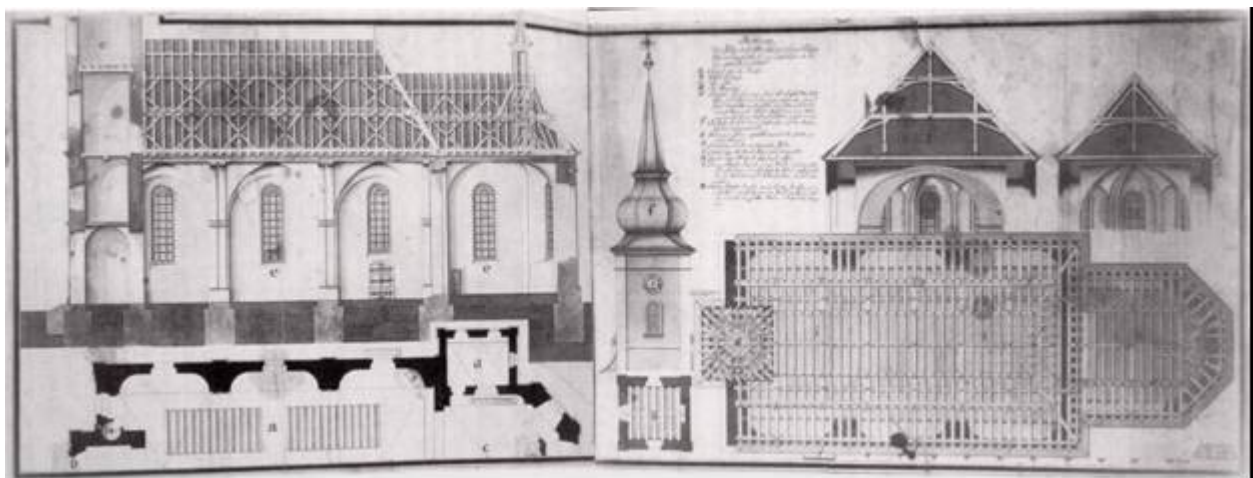
Daher ist während der Ausbildung zum Architekten häufig neben fertigen Plänen auch der Skizzenblock als Teil einer Prüfung mit abzugeben, denn in ihm bildet sich nicht nur ein Endergebnis ab, sondern auch dessen Entstehungsprozess.

Auf dem Roundtable wurde diese Form GDD als „Code des Designers“ bezeichnet, denn wie der Programmcode vom Programmierer verstanden wird, können diese Aufzeichnungen vor allem von Game Designern gelesen werden. - Derzeit kann man sie leider noch nicht kompilieren..

Eine allgemeiner gefasste Funktionsbezeichnung wäre Designnotizen oder auch Designer-Notebook.

1.2 Lasten- / Pflichtenheft: Verbindliche inhaltliche Planung

Dient das Designer-Notebook in erster Linie dem Game Designer selbst, sein spielerisches Konzept zu überprüfen, muss dieses in einem nächsten Schritt auch dem kompletten Entwicklungsteam kommuniziert werden können.



Entwurfsblatt für Dachstuhl, Joseph Klemens Madler 1778

Dazu gehört bestimmt auch, alle Teammitglieder auf das Spiel einzuschwören und eine von allen geteilte Vision zu vermitteln. Ganz nüchtern betrachtet gehört jedoch vor allem dazu, jedem Einzelnen vollständige, präzise und verbindliche Handlungsanweisungen liefern zu können.

Andere Branchen kennen hierzu das Lasten- und das Pflichtenheft, wobei vermutlich kaum irgendwo die Unterscheidung so explizit eingehalten ist, wie sie in der zugehörigen Deutschen Normierung vorgesehen ist.

Gemäß DIN 69905 beschreibt das Lastenheft die vom Auftraggeber festgelegte Gesamtheit der Forderungen an die Lieferungen und Leistungen eines Auftragnehmers innerhalb eines Auftrages. Das Lastenheft beschreibt in der Regel also, was und wofür etwas gemacht werden soll.

Das Pflichtenheft hingegen beschreibt, wie und womit etwas realisiert werden soll. Nach DIN 69905 beinhaltet das Pflichtenheft die vom Auftragnehmer erarbeiteten Realisierungsvorgaben aufgrund der Umsetzung des vom Auftraggeber vorgegebenen Lastenheftes.¹

In der Spielentwicklung begegnen einem häufiger die Begriffe Grobkonzept und Fein- bzw. Detailkonzept. Die Aufgabenteilung zwischen den beiden Dokumenten ist jedoch ähnlich wie bei Lasten- und Pflichtenheft. Der größte Unterschied ist, dass es keine so strikte Trennung zwischen Auftraggeber und Auftragnehmer gibt.

¹ <http://de.wikipedia.org/wiki/Lastenheft>

Welches Dokument auch immer diese Funktion übernimmt, es ist **nicht** der Projektplan. Dieser kann nämlich erst basierend auf den Anforderungen, die sich aus Lasten- und Pflichtenheft ergeben, die zusätzlichen Fragen beantworten **wann und von wem** die aufgestellten Anforderungen bearbeitet werden.

Die quantitativen und qualitativen Anforderungen und die Unterschiede zum Designer-Notebook lassen sich erneut sehr gut anhand des Vergleiches mit der Architektur veranschaulichen.

Es gibt eine breite Palette von unterschiedlichen Plantypen (Zeichnungen), die zur Dokumentation eines Bauvorhabens verwendet werden. Grundrisse, Schnitte und Ansichten gehören nur zu den bekanntesten Vertretern. Und auch diese liegen häufig noch in unterschiedlichen Maßstäben vor, auf denen unterschiedliche Aspekte des Gebäudeentwurfs betont werden.

Hinzu kommen Dachansichten, Fassadenabwicklungen, Fliesenspiegel, zahlreiche Detailzeichnungen sowie noch spezielle Pläne zu Themen wie Brandschutz, Statik, Technischer Ausbau, etc. pp.

Vergleichbar vielfältig dürfte die Aufgabenstellung für ein großes Projekt in der Spielentwicklung sein und jedes Genre hat noch einmal andere Anforderungen, was zu dieser Planung alles dazu gehört. Wieder können Texte, Listen, Tabellen, Grafiken oder Diagramme zur Dokumentation dienen.

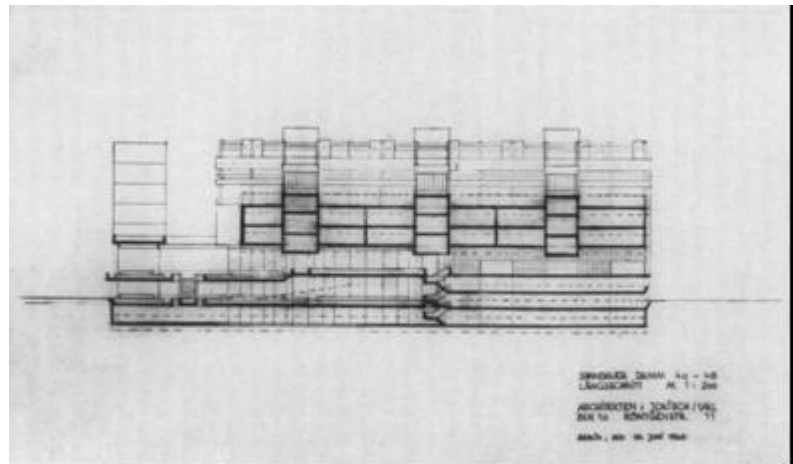
Anders als der „Code des Designers“ dürfen diese Dokumente jedoch keine Frage offen lassen, müssen eindeutig und präzise sein. Sonst steht man im übertragenen Sinne wie die Schildbürger vor einem Haus ohne Fenster oder einer Treppe, die im Nichts endet.

Es wird deutlich, dass dieses Game Design Dokument schlecht alleine vom Game Designer erstellt werden kann, genauso wie der Architekt bei vielen Plantypen auf die Hilfe von Spezialisten des Faches angewiesen ist.

Anders herum ist es riskant, eines dieser Dokumente ganz ohne Beteiligung des Game Designers zu erstellen, da am Ende alle Elemente zusammen greifen müssen zu einem großen Ganzen, das vor allem Spielspaß erzeugen soll.

Und was dem Designer nicht bekannt ist, kann er an anderer Stelle seiner Planung nicht berücksichtigen.

Die beste Lösung scheint daher, Lasten- und Pflichtenheft der Spielentwicklung in enger Kooperation zwischen Game Designer und Projektleiter zu erstellen und dabei bereits die spätere Projektorganisation zu berücksichtigen. Als Beispiel benötigt eine Entwicklung nach Wasserfall-Prinzip eine gänzlich andere Form und Detailtiefe der inhaltlichen Planvorgaben als für einen iterativen Prozess sinnvoll erscheint.



Schnitt zu Wohnhaus, Johannes Uhl 1968

Im Roundtable wurde von vielen Seiten bestätigt, dass diese kreativ-ökonomische Spannung zwischen Game Designer und Projektleiter sich sehr positiv auf das Ergebnis auswirke:

So stünde der Game Designer maßgeblich für das **Was und Wofür**;

Der Projektleiter verantwortet ein meist sehr knappes **Wann und von Wem**

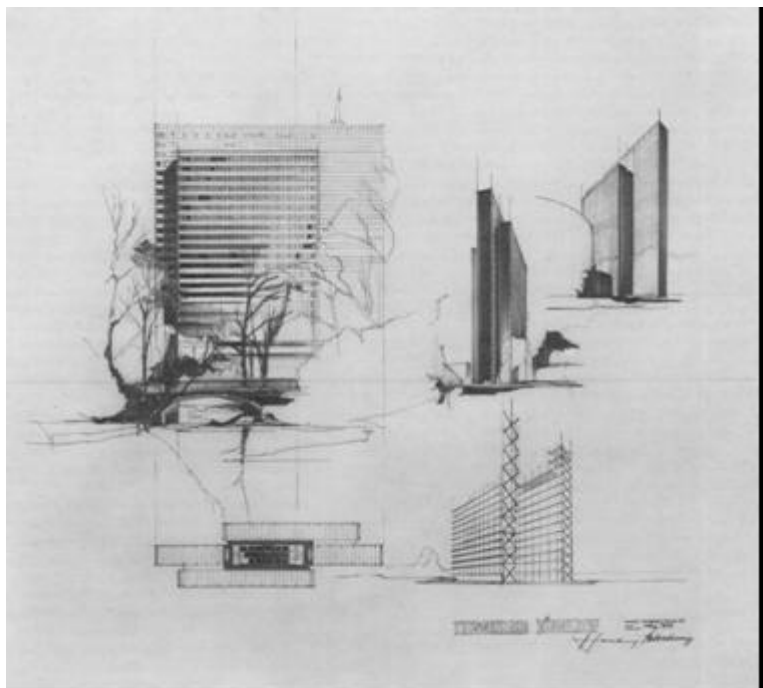
Und über das **Wie und Womit** können beide konstruktiv um die beste Lösung streiten.

Auf die diskutierte äußere Form dieser Dokumente (Word/Wiki, starr/lebendig) gehe ich weiter unten noch einmal kurz ein, pauschal lässt sich hier erst einmal feststellen, dass je nach Team- und Projektablauforganisation das eine oder das andere die Vorteile auf seiner Seite hat.

1.3 Exposé: Pitching und andere Verkaufssituationen

Die größte Verwechslungsgefahr besteht zwischen den GDD als Planungsinstrumente, wie sie im vorigen Kapitel behandelt wurden, und dem GDD als Teil des Pitchings beim Publisher.

Beide werden in der Literatur als Game Design Dokument bezeichnet und doch sind es grundsätzlich unterschiedliche Typen: Jenes ist ein Planungsinstrument, dieses ein Verkaufsinstrument.



Repräsentationszeichnung "Thyssen-Haus",
Hubert Petschnigg / Helmut Hentrich 1979

Beim Pitching geht es darum, den Publisher von vier Dingen zu überzeugen:

- dass man ein cooles Produkt in der Mache hat,
- dieses gut in das Portfolio des Publishers passt,
- die Kosten in einem gesunden Verhältnis zum erwarteten Markterfolg stehen
- und man die Mittel und Fähigkeiten hat, das Produkt zuverlässig zu entwickeln.

Hierbei handelt es sich um ein Verkaufsdokument und dementsprechend sollte es auch aufgebaut und formuliert sein.

Verkaufsdokument heißt dabei nicht, das Blaue vom Himmel zu erzählen oder nur leere Phrasen zu dreschen.

Es heißt zuallererst ein Produkt anzubieten und nicht ein Spiel - die spielerische Qualität ist dabei nur ein Teilaspekt des Produkts. Es heißt ehrlich mit Chancen wie Risiken umzugehen und glaubwürdige Lösungen zum Risikomanagement anzubieten.



Kupferstich Barockpalast, Paul Decker der Ältere 1716

Dies ist umso wichtiger, je weniger Vorleistung bereits erbracht ist an spielbaren Prototypen oder technischen Demos.

Plastisch gesprochen konkurrieren die Projektskizzen bei jedem einzelnen Publisher gegen hunderte andere Projekte von denen am Ende weniger als ein Dutzend Chance auf Realisierung haben.

Das Dokument muss daher in kurzer Zeit viele kritische Fragen beantworten, denn dieser Publisher produziert pro Jahr mehr Titel als der Entwickler und kennt daher jedes typische Risiko ganz genau und will daher wissen, wie damit umgegangen wird.

Anders - und keinesfalls einfacher - ist es noch, wenn ein Verkaufsdokument an jemanden außerhalb der Branche gerichtet ist, zum Beispiel an die regionalen Filmförderer, die zunehmend auch Spieleprojekte fördern. In deren Entscheidungsgremien sitzen derzeit Politiker, Wirtschaftsfachleute, Vertreter von Fernsehanstalten und allenfalls noch Leute vom Film-Fach.

Hier konkurriert ein Projekt mit Projekten aus Film und Fernsehen, zu denen das Entscheidungsgremium einen wesentlich leichteren Zugang hat.

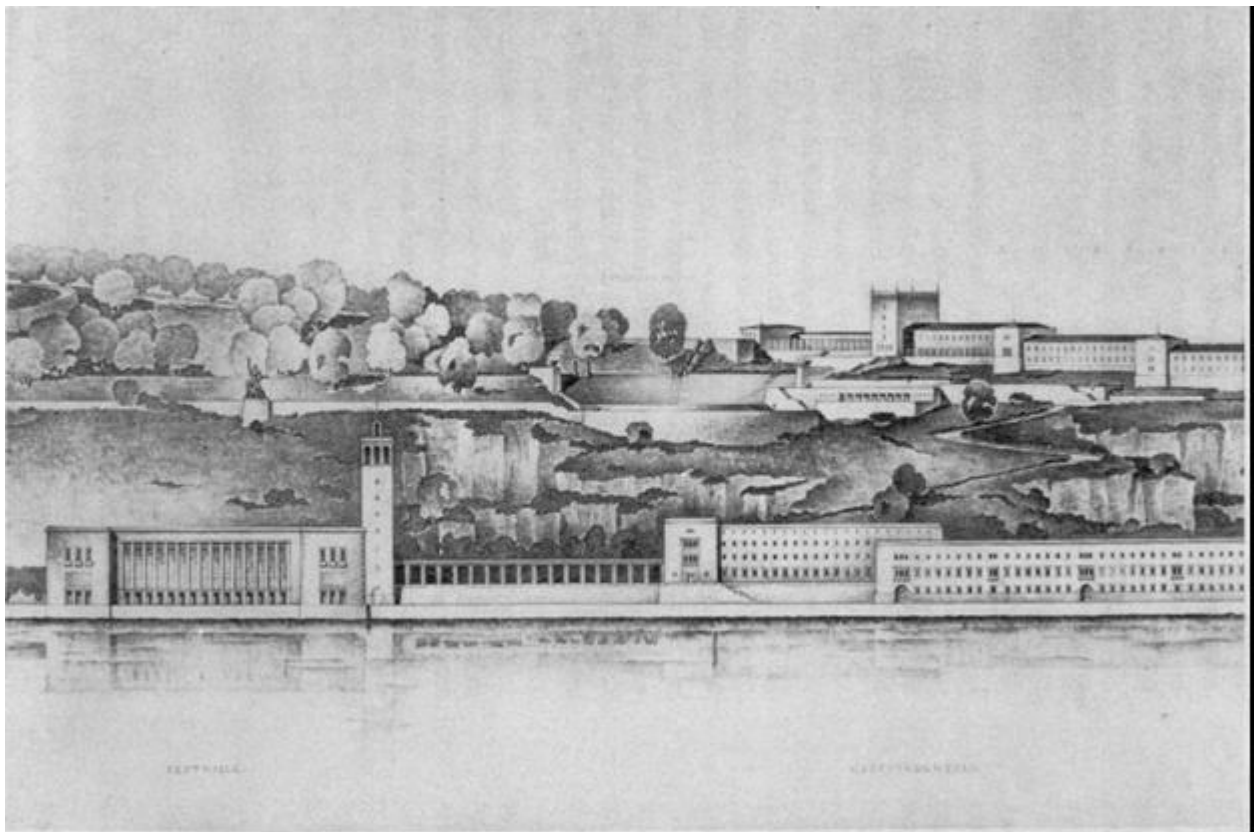
Mehr als bei jedem bisher behandelten Dokumenttyp kommt es darauf an, zielgruppen- und aufgabenorientiert zu schreiben.

Auch hier ist der Blick in Richtung Architektur erhellend: Bei allen größeren Bauprojekten gibt es öffentlich ausgeschriebene Wettbewerbe, an denen zahlreiche Architekturbüros um den Auftrag konkurrieren, indem sie ihre Lösung bereits zu einer ausführlichen Projektskizze ausarbeiten und darin ihre besten Ideen preisgeben.

Für solche Präsentationen wurden perspektivische Zeichnungen perfektioniert und aufwändig gebaute Modelle (Prototypen) sollen versuchen, das Verständnis für den eigenen Entwurf zu verbessern. Visualisierung von Architektur arbeitet inzwischen ebenso selbstverständlich mit 3D-Renderings (und zunehmend Echtzeitmodellen) wie die Computerspielindustrie: Viele Grafiker arbeiten für beide Branchen.

Große Architekturbüros unterhalten indes ganze Abteilungen von Spezialisten, die allein auf diese Wettbewerbe hinarbeiten..

Für diese Art von Dokument möchte ich den Begriff Exposé verwenden, der auch in anderen Branchen für so eine Projektskizze Verwendung findet.



Wettbewerbs-Entwurf Hochschulstadt Bratislava, Willi Kreuer 1941

Und weil die Tatsache, dass es bei Ascaron einen dezidierten Leitfaden für solche Exposés gibt, während des Roundtables so belächelt wurde, möchte ich empfehlen, ihn vor dem Hintergrund der Aufgabe eines solchen Exposés noch einmal genauer anzugucken. Alternativ oder besser ergänzend kann man natürlich den/die Publisher des eigenen Vertrauens um seine entsprechende Pitching-Anforderung ersuchen.

1.4 Formales: Final oder lebendig, Word oder Wiki

Unter formalen Aspekten möchte ich all das verstehen, was die äußere Form sowie die Struktur des Dokuments betrifft und welcher Art der Nutzung das Dokument unterliegt.

Natürlich hat die Form auch Einfluss auf den Inhalt. Deshalb sollte diese Frage nicht als Nebensächlichkeit oder ‚Formalität‘ abgetan werden. Die Intensität der Diskussion um diese Themen zeugte davon, wie Erfahrungen der Vergangenheit des Einzelnen auch dessen Einstellung zu formalen Aspekten geprägt hat.

Da wäre zum Beispiel die Frage, ob ein GDD vor Produktionsbeginn einen finalen Status erreicht oder auch darüber hinaus lebendig bleibt.

Zunächst einmal stellt eigentlich niemand einen Plan auf mit der festen Absicht, ihn später noch einmal zu ändern. Wenn es zu Änderungswünschen kommt, dann entweder, weil

1. diese von Außen an das Projekt getragen werden, zum Beispiel durch Marketing, Publisher, Presse oder Geschäftsführer,
2. während der Durchführung neue Erkenntnisse entstanden sind, wie das Ergebnis verbessert werden kann, oder
3. sich die ursprüngliche Planung nicht aufrecht halten lässt, zum Beispiel weil Fehler im Plan entdeckt wurden oder einzelne Aufgaben sich als aufwändiger herausstellen als geplant.

In Fall 1 kann man versuchen, die Änderungswünsche abzublocken, in Fall 2 ebenfalls - wobei dabei die Teammotivation leiden kann, aber spätestens in Fall 3 kommt man um eine Planänderung nicht umhin, will man nicht sehenden Auges ins Verderben gehen.

An diese Stelle gehört nun eine sehr intensive Diskussion um die beste Projektorganisation für die Spieleentwicklung: Als Gegenpole seien auf der einen Seite das Wasserfallmodell (*Planung wird vollständig finalisiert, bevor mit der Produktion begonnen wird*) genannt und auf der anderen Seite Agile Entwicklungsmethoden (*der Plan wird mit Fortschritt des Projekts gerade so weit konkretisiert, dass die zügige Entwicklung nicht behindert wird*).

Diese Diskussion würde jedoch den Rahmen dieses Artikels sprengen und hängt auch von zu vielen Faktoren wie Studioorganisation oder Vertragsgestaltung ab, als das ich mir eine allgemein gültige Empfehlung anmaßen würde.

Ich hege die Vermutung, dass viele aus Sorge vor dem gefürchteten Feature Creep lieber auf eine finalisierte Planung setzen. Diesen kann ich nur zwei Dinge ans Herz legen. Sich zum einen doch einmal mit Agilen Methoden zu beschäftigen und wie sie gerade das Problem Feature Creep elegant lösen (wenn man sie konsequent einsetzt).

Zum anderen möchte ich dringend anraten, die eigenen GDD zumindest formal so zu gestalten, dass sie während der Produktion geändert und diese Änderung allen Betroffenen kommuniziert werden können.

Auch in die sorgfältigste Planung können sich Fehler einschleichen, die zwingend beseitigt werden müssen und wenn es dafür keinen bewährten Prozess gibt, kann das folgende Kommunikationschaos Folgefehler produzieren.

Wer in Richtung Agilen Methoden oder lebendige GDD denkt, muss sich ohnehin intensiv mit leichter Zugänglichkeit der Dokumente, Versionierung derselben und Benachrichtungen bei Änderungen beschäftigen.

Damit wäre auch das letzte Thema des Roundtable und damit dieses Artikels erreicht: Word oder Wiki?

So hitzig die Diskussion darum war, so wenige Argumente wurden für das eine oder andere genannt. Dies möchte ich versuchen nachzuholen, da ich mit beidem lange und intensiv gearbeitet habe und noch immer arbeite.

Ein Wiki hat seine unbestreitbaren Stärken, wenn es um lebendige Dokumente geht. Es bietet leichten Zugang für alle Beteiligten, die immer den neuesten Stand sehen und dank Versionierung auch Änderungen hervorheben können. Gute Wikis bieten auch Funktionen, um über Änderungen an abonnierten Themen per E-Mail informiert zu werden, so dass bei richtiger Benutzung niemand eine ihn betreffende Änderung verpasst. Die Struktur einzelner Themenseiten erlaubt es, die Gesamtplanung auf den

Bedarf derjenigen zuzuschneiden, die sie später umsetzen und ihnen dank Verlinkung dennoch einfachen Zugang zu benachbarten Themen anzubieten.

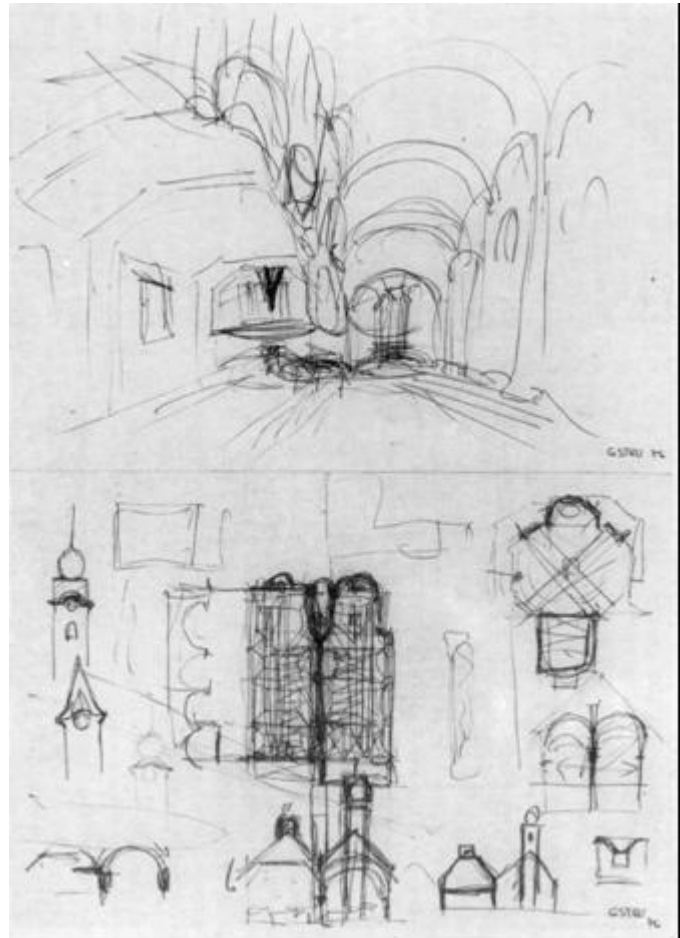
Diese netzartige Struktur birgt jedoch auch die Gefahr, dass zu viel oder - noch schlimmer - zu wenig Information abgerufen wird, weil Unsicherheit darüber herrscht, welchem Link denn womöglich noch gefolgt werden muss, um alle benötigten Informationen zusammen zu haben.

Schwächen in der vom Autor angelegten Struktur wirken sich gravierender aus als bei klassischen linearen Texten.

Die zweite große Schwäche von Wikis liegt darin, dass in den vergleichsweise schwachen Editierfunktionen bieten. Zwar verbreiten sich WYSIWYG-Editoren langsam, aber auch diese sind Jahre hinter ausgereiften Texteditoren hinterher und unterstützen meist nur Basisfunktionen.

Man muss einfach feststellen, dass Word ein unglaublich mächtiger Texteditor ist, der jedem Wiki um Jahre voraus ist. Ich nenne nur die flexiblen Formatvorlagen, die Gliederungsansicht, 'Änderungen nachverfolgen' und das Handling verknüpfter Medien von Bildern bis hin zu Tabellen.

Natürlich hängt das gewählte Tool auch vom Dokumenttyp ab. Für ein Exposé zum Beispiel wäre ein Wiki gänzlich ungeeignet - da würde eher noch Powerpoint auf den Plan treten.



**Skizze Kirchenerweiterung Illmitz,
Johann Georg Gsteu 1976**

1.5 Resümee

Game Design Dokumente sind genauso vielfältig, wie sie sich in den Diskussionsbeiträgen des Roundtable dargestellt haben. Dabei hat jede Dokumentart seine eigenen Regeln, die sich aus dem jeweils zu lösenden Problem ableiten.

Jede formale Lösung hat ihre Stärken und Schwächen, für die man bewusst entscheiden kann.

Kehren wir zurück zu der Grundfrage nach dem „Sinn oder Unsinn von Game Design Dokumenten“. Es ist mir auch überhaupt nicht klar, welchen „Unsinn“ eine schriftliche Dokumentation bei der Planung so komplexer Gegenstände wie Computerspielen haben könnte - und welche Alternative es gäbe.

Eine zukünftige Diskussion über Game Design Dokumente sollte sich daher von vornherein gezielt darauf fokussieren, deren Sinn gezielt zu mehren.

Dabei fallen mir spontan drei wichtige Problemfelder ein:

- Früherkennung spielerischer Probleme: Wie können GDD noch vor der Erstellung spielbarer Prototypen Fehler im Spielsystem vermeiden helfen? (Stichworte: Papier-Prototypen, Modellrechnungen, Peer-Reviews)
- Vermeidung von Fehlern aus Missverständnissen: Wie können GDD so geschrieben werden, dass sie auch so gelesen werden, wie sie gemeint waren? (Stichwort: Problem- und Zielgruppenorientiertes Schreiben)
- Änderungen Planen und Dokumentieren: Welche Schwierigkeiten ergeben sich für GDD, wenn sie - unverschuldet - mit häufigen Änderungen umgehen müssen? (Stichworte: Reaktion auf den Markt, Iterative Projektentwicklung)

Text ©2006 Björn Lilleike

Alle Abbildungen aus:
Architektenzeichnungen 1479-1979,
Verlag Volker Spiess, Berlin 1979