

Claus Pias

## Adventures als Ereignis und Zählung: Die ersten Adventurespiele im Zeichen ihrer Topologie und Datenbank-Ontologie

2002

<https://doi.org/10.25969/mediarep/17529>

Veröffentlichungsversion / published version  
Zeitschriftenartikel / journal article

### Empfohlene Zitierung / Suggested Citation:

Pias, Claus: Adventures als Ereignis und Zählung: Die ersten Adventurespiele im Zeichen ihrer Topologie und Datenbank-Ontologie. In: *Dichtung Digital. Journal für Kunst und Kultur digitaler Medien*. Nr. 22, Jg. 4 (2002), Nr. 2, S. 1–25. DOI: <https://doi.org/10.25969/mediarep/17529>.

### Nutzungsbedingungen:

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

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

### Terms of use:

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

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

# Adventures als Ereignis und Zählung: Die ersten Adventurespiele im Zeichen ihrer Topologie und Datenbank- Ontologie

Von Claus Pias

Nr. 22 – 15.03.2002

## Abstract

Aller Rede vom offenen Kunstwerk, interaktiver Literatur oder Cinematisierung zum Trotz umschreiben Adventurespiele nach wie vor jene Probleme von Graphentheorie und Datenbankorganisation, deren historischer Bastard sie sind. Wenn in Adventurespielen Entscheidungsbäume figurieren, wenn es um Wege und Verknüpfungen innerhalb einer positionalen Logik von Datensätzen geht, dann fehlt Adventures die für Erzählungen konstitutive Zeitlichkeit und sie unterlaufen die von Lessing behauptete mediale Differenz ästhetischer Zeichenregime. Der Text beschäftigt sich folglich mit den ersten Adventurespielen, stellt ihre Datenbank-Ontologie und Topologie vor und fragt, was unter diesen Bedingungen als Ereignis zählt oder erzählt.

## 1. Höhle, Spiel und message processing<sup>1</sup>

Um Jahr 1838, also genau 130 Jahre, bevor die *ARPA* bei *Bolt, Beranek & Newman* die Entwicklung der ersten vier *Internet Message Processors (IMPs)* in Auftrag gibt, die William Crowther, Frank Heart, Robert Kahn, Severo Ornstein und David Walden 1970 präsentieren werden,<sup>2</sup> tätigt Frank Gorin einen für die Geschichte der Adventurespiele folgenreichen Grundstückskauf.<sup>3</sup> Er erwirbt die sogenannte *Mammoth Cave*, die mit über 350 Meilen größte Höhle der Welt im Karst von Kentucky. Eben jenem Frank Gorin gehörte auch der Sklave Stephen Bishop, und es ist wohl der tuberkuloseinduzierten Philanthropie seines Herrn geschuldet, daß Bishop nicht nur in Latein und Griechisch, sondern auch in Speläologie dilettieren konnte. So erforschte Bishop die Höhlen seines bettlägerigen Herrn und verdoppelte das kartographierte Territorium schon innerhalb eines Jahres. Das unterirdische

Dunkel durch Benennung lichtend, entstand nicht nur eine (halb klassische, halb folkloristische) Topologie mit Raumbezeichnungen wie *River Styx*, *Snowball Room*, *Little Bat Avenue* oder *Giant Dome*, sondern auch ein Inventar von Merkwürdigkeiten wie blinden Fischen, stummen Grillen, Fledermäusen und indianischen Artefakten. Bishop starb 1856, doch seine 1842 aus dem Gedächtnis gezeichnete Karte blieb bis in die 1880er Jahre in Gebrauch.



Abb.1: Das Original des „Giant Dome“ in der Mammoth Cave, Kentucky

John Croghan, der nächstfolgende Besitzer, baute die Höhlen zum touristischen Ausflugsziel aus. Die Nachbarorte ruhten nicht, und die Eröffnung der *Great Onyx Cave* im nahegelegenen Hochins Valley führte in den 20ern zu den sog. *Kentucky Cave Wars*. Falsche Polizisten, orientierungslose Führer und brennende Ticket-Services bestimmten die Lage, so daß die Höhlen 1941 für die Öffentlichkeit geschlossen wurden. Nach dem Krieg wurden sie mit einiger Verspätung als Nationalpark wiedereröffnet, und Höhlenforscher suchten nach einem lange vermuteten Verbindungsgang zwischen *Flint Ridge* und *Mammoth Cave*. Der Durchbruch gelang 1972 einer jungen Physikerin namens Patricia Crowther, die – ausgehend von einem Raum namens *Tight Spot* – jene entscheidende "muddy passage" auf der Karte verzeichnen konnte, durch die man zur *Mammoth Cave* gelangt.



Abb.2: Das Original des „Muddy Passage“ in der Mammoth Cave, Kentucky

Patricia war nun niemand anderes als die Ehefrau jenes eingangs erwähnten William Crowther, des genialen Programmierers der *Internet Message Processing Group*, der seine militärisch-wissenschaftlichen Routing-Probleme bei *BBN* ab und an ruhen ließ, um die speläologischen seiner Gattin auf den dortigen Dienstrechnern zu erfassen und zur Weitergabe an die *Cave Research Foundation* aufzubereiten. Und (s)einer Scheidung ist es zu danken, daß Crowther nicht nur die ausbleibenden Höhlen-Ausflüge auf seinem Rechner zu simulieren suchte, sondern ab 1973 diese auch in einer kinder- und laienfreundlichen Form implementierte.<sup>4</sup> Die erste Verion – auch unter dem treffenden Namen *Colossal Cave* bekannt – basierte, in schönem Angedenken an die Lieblingsregion des Pärchens, auf einem vier Ebenen tiefen Modell der *Bedquilt Cave* und übernahm folgerichtig auch Raumbezeichnungen wie *Orange River Rock Room* aus der realen Höhlensituation. Diese – aus *FORTRAN*-Beschränkungen in schönem Doppelsinn "ADVENT" genannte – Begründung eines neuen Spieltypus kursierte im jungen Netz und fand ihren kurzen Weg zu Don Woods vom *Stanford Artificial Intelligence Lab (SAIL)*. Nachdem Crowther 1976 zu *XEROX PARC* gewechselt war, erweiterten er und Woods die Karte des Programms, überzogen sie mit Ortsbeschreibungen in Tolkien'schem Ambiente und setzten dem Herumlaufen ein Ende. In Form von aufzufindenden Schätzen und Objekten gab es nun eine Geschichte mit einem Ende namens maximale Punktzahl.

Die Semantisierung eines kartographischen Settings, seine Anreicherung durch Gegenstände und Rätsel war für die geübten Spieler des papierbasierten Rollenspiels *Dungeons and Dragons* nichts ungewöhnliches. Erstaunlich ist jedoch, wie nachdrücklich sich die vom Realen abgenommene Topologie dieses ersten

Adventure ins Raumgedächtnis seiner Spieler einschrieb, die sich bei einem Besuch der *Mammoth Cave* sofort und ohne Karte darin zurechtfinden.<sup>5</sup> Die Orientierungsleistung, die das Modell ADVENT abfordert, erweist in Anwesenheit des Modellierten als gelungene Programmierung seiner Spieler. Kurt Lewins topologische Psychologie, die ja zu der Einsicht führte, daß wir in Räumen immer schon auf bestimmten Wegen laufen ohne uns diese Wege eigentlich klarzumachen und ohne sie selber im strengen Sinne gebahnt zu haben, bekommt hier einen ganz neuen Sinn. Wenn Derrida davon spricht, daß die Architektur als eine "Schrift des Raumes" das Ich "instruiert", dann gebührt diese Leistung hier einer nicht metaphorisch, sondern ganz wörtlich zu verstehenden Schrift eines Textadventures.<sup>6</sup> Das wiederholte Tippen von Befehlen wie "LEFT", "RIGHT" oder "UP" als Kommandos an einen Avatar im symbolischen Spielraum hallt im Realen zurück als Kommando an den eigenen Körper. Der Gang des Spielers durch die Höhle ist kein Heideggersches "Bewegen" mehr, das den Raum erst erschließt, sondern tatsächlich nur noch "Bewegen" in einem déjà-vu vorhandener Trassen in bereits gebahnten Räumen.



Abb.3: Die Internet Message Processing Group bei BBN  
(Will Crowther rechts, kniend)

Jenseits von realweltlichen Höhlen und Fantasyliteratur, von Rollenspielen und unglücklichen Ehen trug jedoch ein technisches Problem entscheidend zur Emergenz von Adventurespielen bei. Crowther gehörte nämlich zum Programmiererteam der *IMP-group*, die sich aus echtzeiterfahrenen Ingenieuren

des *Lincoln Lab* rekrutierte und deren wesentliche Aufgabe es war, die Routing-Probleme des *ARPAnet* zu lösen.<sup>7</sup> Seine erste Aufgabe resultierte aus den *ARPA*-Spezifikationen, die ein dynamische Routing verlangten: fällt eine Verbindung zwischen zwei Punkten des Netzes aus, so muß der Datenfluß über eine (möglichst geringe) Zahl anderer *nodes* zum Ziel finden. Die Nachricht wurde folglich in *packets* zerlegt, die jeweils nur eine Absender- und eine Empfängeradresse, nicht aber eine Wegbeschreibung bei sich führen, und der Rest blieb der postalischen Intelligenz des Systems überlassen. Das zweite Problem hieß Transparenz und meinte die Verbergung der gesamten Subnet- und Routing-Entscheidungen vor dem Benutzer. Da Post das ist, was keine Adresse hat, durfte bei einer Verbindung von der *UCLA* nach Utah eben auch nur Utah erscheinen und nicht etwa der Weg dorthin. Transparenz bestand also gerade nicht in der Sichtbarmachung von Wegen, sondern in deren Verschwinden, darin, daß Verwaltungsentscheidungen nicht mehr nachvollzogen werden können.

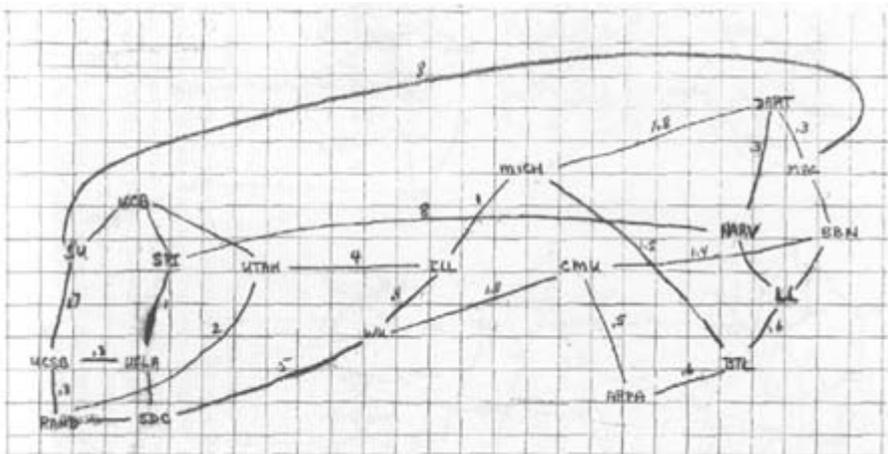


Abb.4: Routing-Entwurf des ARPAnet

Aus dieser Ausgangskonfiguration des ersten Adventurespiels lassen sich mindestens drei Schlüsse ziehen. *Erstens* basieren Adventurespiele auf *Karten* oder genauer: auf Orten und Wegzusammenhängen. Diese Orte können "Räume" einer Höhle sein oder *nodes* eines Netzes, wobei die Wege zwischen ihnen auf eine je spezifische Weise verschwinden. *Zweitens* sind Adventurespiele Geschichten in jenem basalen Sinn, daß sie einen Anfang und ein Ende haben. Dies können zwei postalische Adressen sein oder Auszug und Heimkehr eines Helden oder Start- und Haltepunkte eines Programms. *Drittens* sind Adventures Serien von Entscheidungen, die auf Orte der Karte verteilt sind. Dies können Rauten in einem Flußdiagramm sein, Knoten in einem Netzwerk oder 'Scheidewege' eines Helden.

Im folgenden möchte ich etwas genauer auf den Zusammenhang zwischen Erzählung und Topographie in Adventurespielen eingehen und auf ihr mögliches Aufgehen (oder Nichtaufgehen) im Rahmen einer mathematischen Graphentheorie hinweisen.

## 2. Erzählung

Crowthers und Woods *Adventure* beginnt für seinen Spieler mit den berühmten Sätzen:

You are standing at the end of a road before a small brick building. Around you is a forest. A small stream flows out of the building and down a gully.

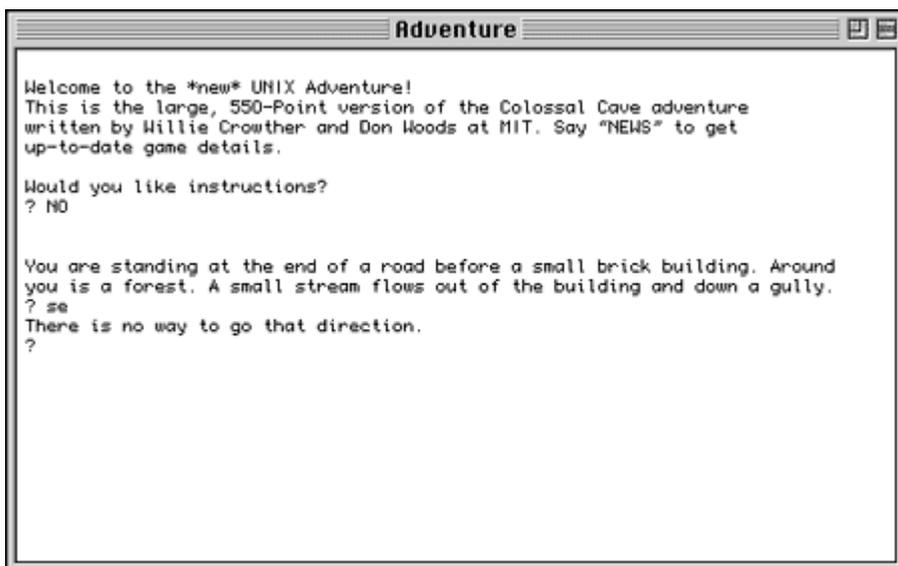


Abb.5: Startup-Screen des Adventures von Will Crowther und Don Woods

Zugleich (und auf Computerseite) beginnt es mit ganz anderen Sätzen, nämlich jenen unsichtbaren Instruktionen, die die Möglichkeitsbedingung der Spieler-Instruktion darstellen. Diese eignen sich allerdings weniger dazu, (vor-)gelesen, als vielmehr dazu, von Compilern übersetzt zu werden. Was dort aufgeschrieben ist, ist – in Anlehnung an Wolfgang Hagen – kein Spiel, ohne daß es übersetzt würde in die Physikalität von Hardwarezuständen, womit 'res gestae' und 'historia rerum

gestarum' zusammenfallen.<sup>8</sup> Wenn Adventurespiele allseits unter eine Literatur names "Interactive Fiction" subsumiert werden, dann unterschlägt diese Beschreibung einen unsichtbar gewordenen Text namens Programmcode, der die Möglichkeitsbedingung der 'gegenständlichen Schicht des literarischen Werkes' (Roman Ingarden) ist.<sup>9</sup> Wo das Eintauchen einer *madeleine* in Lindenblütentee stillschweigend und lebensweltlich voraussetzen kann, daß es nicht nur *madeleines* und Tee gibt, sondern auch Hände, die Gebäck greifen können, Tassen, in die man Tee gießen kann, daß Tee flüssig und *madeleines* kompatibel mit Tassendurchmessern sind, da bedürfen Adventurespiele erst einer ontologischen Klärung dessen, was 'der Fall ist'. "Interactive Fiction" muß also gerade umgekehrt gelesen werden: als Fiktion der Interaktivität, zu der es des programmiertechnischen Entwurfs eines Weltzusammenhangs bedarf.

Demnach beginnt *Adventure* erst einmal, also im *FORTRAN*-Sourcecode, mit den Zeilen:

```
DIMENSION LINES(9650)
DIMENSION TRAVEL(750)
DIMENSION KTAB(300),ATAB(300)
DIMENSION LTEXT(150),STEXT(150),KEY(150),COND(150),ABB(150),
1 ATLOC(150)
DIMENSION PLAC(100),PLACE(100),FIXD(100),FIXED(100),LINK(200),
1 PTEXT(100),PROP(100)
DIMENSION ACTSPK(35)
DIMENSION RTEXT(205)
DIMENSION CTEXT(12),CVAL(12)
DIMENSION HINTLC(20),HINTED(20),HINTS(20,4)
DIMENSION MTEXT(35)
```

Es handelt sich – wie der Befehl so treffend sagt – um die Dimensionierung der Variablen einer Welt, die durch (jeweils maximal) 9650 Wörter beschreibbar ist, 150 unterscheidbare Orte hat, in der 100 Objekte vorkommen, in der 35 bestimmte Handlungen möglich sind und in der eine Sprache gesprochen wird, die aus 300 Wörtern besteht.<sup>10</sup> Die Belegung dieser Variablen erfolgt durch das Auslesen einer externen Datenbank, so etwa die Beschreibung der Eröffnungssequenz:

```
1 YOU ARE STANDING AT THE END OF A ROAD BEFORE A SMALL BRICK
BUILDING.
1 AROUND YOU IS A FOREST. A SMALL STREAM FLOWS OUT OF THE
BUILDING AND
1 DOWN A GULLY.
```

Und erst wenn alle Variablen belegt sind, beginnt das Spiel mit der Ausgabe des ersten (Daten-)Satzes (vom Typ "LTEXT"), und das Programm verwaltet fortan nur noch, was zu bestimmten Zeiten an bestimmten Orten mit bestimmten Objekten

unter welchen Vorbedingungen möglich ist. Adventurespiele – so könnte man vorgreifend sagen – bestehen also aus *records* oder Datensätzen und deren Verwaltungsrichtlinien, nach denen der Spielverlauf prozessiert (wird). Um das Spiel spielbar zu machen, bedarf es der Simulation einer konsistenten Welt, die in entscheidenden Parametern der Lebenswelt gleicht: Briefkästen können sich öffnen, Fensterscheiben sind durchsichtig, Flaschen enthalten Wasser. Was daher programmtechnisch auf dem Arbeitsplan steht, ist ein *Reverse Engineering* aristotelischer Kategorienlehre, heißt es doch dort (Kapitel 4, 1b25-2a3):

Von den Dingen, die ohne Verbindung ausgesagt werden, bezeichnet jedes eine Substanz oder eine Quantität oder eine Qualität oder ein Relativum oder einen Ort oder eine Stelle in der Zeit oder eine Lage oder das Haben von etwas oder ein Tun oder Erleiden. Substanz ist -um eine ungefähre Vorstellung zu vermitteln- so etwas wie Mensch, Pferd; Quantität so etwas wie zwei Ellen lang, drei Ellen lang; Qualität so etwas wie weiß, schriftkundig; ein Relativum so etwas wie doppelt, halb oder größer; ein Ort so etwas wie im Lyzeum, auf dem Marktplatz; eine Stelle in der Zeit so etwas wie gestern oder letztes Jahr, eine Lage so etwas wie liegt, sitzt; ein Haben so etwas wie hat Schuhe an, ist bewaffnet; ein Tun so etwas wie schneiden, brennen; ein Erleiden so etwas wie geschnitten werden, gebrannt werden.

Wo Aristoteles sich zunächst nur gegen die Mehrdeutigkeit des Prädikats "sein" in der platonischen Ontologie wendet und versucht, *verschiedene* Weisen der Prädikation zu unterscheiden, erscheint später die Hoffnung, zehn Kategorien könnten ausreichen, um *alle* verschiedenen Funktionen von "ist" zu erfassen. Die Problemorientierung auf Vollständigkeitsbeweise verfehlt jedoch die Kategorienlehre von Computerspielen völlig. Bei ihnen geht es nicht um die vollständige Klassifizierung eines mannigfaltig Vorhandenen, sondern um die Möglichkeitsbedingung alles Seienden in der künstlichen Welt selbst. In einer Kategorientafel würden sie den "reinen Verstandesbegriffen" näherücken: denn wie der Verstand nur durch sie ein Objekt denken kann, so kann im Programm nur durch sie ein Objekt überhaupt sein.<sup>11</sup>

Nachdem die Welt dimensioniert ist, halten die Dinge Einzug. Als "objects" haben sie alle (Held und Feinde, Waffen und Schätze, Briefkästen und Fensterscheiben) den gleichen Status, weshalb in einer gott- wie programmiererdurchwalteten Welt auch alles mit allem verwandt ist. "End of a Road" ist im Beispiel das elterliche Objekt (*parent*) von "you", "gully" und "small brick building" (*children*), die untereinander Geschwister (*siblings*) sind. Diese Pointer-Struktur bedeutet gewissermaßen die Abschaffung transzendentaler Obdachlosigkeit, denn logischerweise kann nichts nirgendwo sein. Als hierarchisches Inventar aufgeschrieben liest sich dies:

[ 41] " "  
. [ 68] "End of Road"  
.. [ 21] "you"  
.. [239] "small brick building"  
.. [127] "gully"

wobei das [41] ein programmiertechnisches Dummy ist, das nirgends sichtbar aber zugleich überall ist, weil es die logische Ermöglichung aller anderen Objekte ist, die auf der hierarchischen Stufe von beispielsweise Objekt [68] liegen. Mit Lacan gesprochen ist ja jedes Objekt das, was die Leere seiner eigenen Stelle ausfüllt, einer Stelle, die dem ausfüllenden Objekt vorangeht, so daß [41] vielleicht ein programmtechnisches "objet petit a" heißen könnte.

Wie auch immer: Jedes Objekt hat Attribute und Eigenschaften. Erstere sind simple *flags*, die nur gesetzt oder ungesetzt sein können. Eine aristotelische Qualität wie "brennbar" bedeutet also nur, daß unter den 32 verwalteten Status-bits eines für Brennbarkeit (*burnability*) steht und gesetzt ist. Eigenschaften hingegen haben numerische Werte, die beispielsweise Adressen von Strings sein können. Zu den Eigenschaften gehören die Namen der Objekte, ihre Größe oder ihr Gewicht, aber auch die Adressen von speziellen Routinen oder ihre Punktzahl bei erfolgreicher Benutzung.

Daß die Welt des Spiels notwendigerweise eine relationale Datenbank ist, hat — schon aus Gründen der Endlichkeit von Speichern — zur Folge, daß das, was keinen Datensatz hat, auch nicht existiert. Diese schlichte Einsicht ist jedoch entscheidend für den Zusammenhang von Literatur und Spiel. Was nämlich in den Strings der Raumbeschreibungen als Literatur steht, muß noch lange nicht in der Datenbank, auf die das Spiel aufsetzt, vorhanden sein. Beispiel: "To the north a narrow path winds through the trees": der Weg ist gangbar, doch die Bäume lassen sich nicht fällen, weil sie keine Objekte sind und folglich nicht zur Disposition stehen. Die spielbare Welt des Adventures ist nicht an sich, sondern immer schon für uns. Was zur Welt der Zwischentexte gehört und nur halluzinierbar ist, nennt sich Literatur, was zur Welt der Objektdatenbank gehört und referenzierbar ist, nennt sich Spiel. Oder genauer: nicht alle Wörter in den Texten adressieren Objekte, aber spielbar ist nur, was eine Adresse hat. Spielen heißt folglich: nehmen, was auf seine (Wahr)nehmung wartet und wahrgenommen als (Ver-)Handelbares (oder als Ware) genommen wird.

Dieses Handeln geschieht in Textadventures auf Kommandozeilenebene durch die Eingabe von Sätzen wie "GO NORTH", "OPEN GULLY" usw., die von einem Parser gemäß der Generativen Transformationsgrammatik analysiert und anhand von "verb frames" oder "stereotypes" verarbeitet werden. Die Verstehensillusion, die sie erzeugen, beruht darauf, daß sie überlieferte Sprache in verarbeitbare technische Sprache zu konvertieren vermögen.<sup>12</sup> Oder umgekehrt: Daß sie eine formale

Sprache, eine Befehlssprache im Kommandozeilen-Format in Lexik und Syntax so zu gestalten vermögen, daß sie dem lexikalischen und syntaktischen Format überlieferter Sprache ähnlich erscheint. Es erübrigt sich zu zeigen, daß jedes Wort eindeutig definiert sein muß, daß die Syntax der Eingaben korrekt sein muß, damit das Spiel spielbar ist und daß der Spieler sich nur mit den Wörtern zu Wort melden kann, die ihm das Programm erlaubt. Bemerkenswert ist vielmehr, daß das Stellende der Vorschrift beim Schreiben in Kommandozeilen zugleich das Herstellen eines Spielverlaufs ermöglicht, also als *Poiesis* gewissermaßen das Anwesende in die Unverborgenheit hervorkommenläßt. Am Interface des Textadventures wird technische Sprache zum Medium der Entbergung.

Die Freiheitsillusion des Adventures besteht darin, daß es diese Grenze von Literatur und Datenbank verwischt, oder anders: daß es nur zur Wahrnehmung dessen instruiert, was auch Objekt ist:

A table seems to have been used recently for the preparation of food. A passage leads to the west and a dark staircase can be seen leading upward. A dark chimney leads down and to the east is a small window which is open. On the table is an elongated brown sack, smelling of hot peppers. A bottle is sitting on the table. The glass bottle contains: A quantity of water.

Während das leise Lesen bildgebender Literatur einem alphabetisierten, romantischen Publikum den Boden der Innerlichkeit bereitete, wird das Lesen von Adventurespielen zur Lektüre von 'Informanten'. Die Geschichte des Spielverlaufs als Folge von Tableaus dinglicher Rätsel besteht gewissermaßen im Abtasten von *keywords* und der Rekonstruktion einer verlorengegangenen Gebrauchsanweisung. Adventurespiele treffen sich in dieser Veräußerlichung mit der Poetologie des *Nouveau Roman*, beispielsweise derjenigen Robbe-Grilletts, dessen Texte wie Beschreibungen von Adventure-Räumen gelesen werden können, z.B. diese Stelle aus *Der Augenzeuge*:

Es gab dort also, vom Fenster aus linksherum gesehen [...]: einen Stuhl, einen zweiten Stuhl, den Toilettentisch (in der Ecke), einen Schrank, einen zweiten Schrank (der bis in die zweite Ecke reichte), einen dritten Stuhl, das mit seiner Längsseite an der Wand stehende Bett aus Vogelkirschbaumholz, einen sehr kleinen, runden, einfüßigen Tisch mit einem vierten Stuhl davor, eine Kommode (in der dritten Ecke), die Tür zum Flur, eine Art Schreibschrank, dessen Platte hochgeklappt war, und schließlich einen dritten Schrank, der schräg in der vierten Ecke stand, vor dem fünften und sechsten Stuhl. In diesem letzten wuchtigsten und immer abgeschlossenen Schrank befand sich auf dem unteren Brett in der rechten Ecke die Schuhschachtel, in der er seine Schnur- und Kordelsammlung unterbrachte.<sup>13</sup>

Die Dinge im Hotelzimmer des Uhrenvertreters Matthias könnten wohl leicht in einem Objektbaum verzeichnet werden, hätten Attribute wie "containability",

verzeichneten Enthaltene als *properties* (dritter Schrank > Schachtel > Schnüre) oder würden durch solche disambiguiert (erster, zweiter, dritter ... Stuhl).

Ich erspare mir eine Diskussion des Begriffs vom "offenen Kunstwerk", nicht ohne zu konstatieren, daß Adventurespiele keine offenen Kunstwerke sind. Stattdessen verweise ich exemplarisch auf ein funktionales Modell von Erzählung, wie es Roland Barthes bereitstellt.<sup>14</sup> Dessen strukturelle Analyse zerlegt Erzählungen in Einheiten, und weil diese Segmente einen funktionalen Charakter haben, entsteht aus ihnen eine Geschichte oder 'Sinn' (und zwar in der Bedeutung einer Richtung von Beziehungen). Bei den funktionalen (oder "distributionellen") Elementen unterscheidet Barthes die "Kardinalfunktionen" (oder: "Kerne") von den "Katalysen". Kerne eröffnen eine für die Geschichte folgetragende Alternative: Ein Gullydeckel kann geöffnet oder nicht geöffnet werden, daß er aber überhaupt da ist, eröffnet (wie der Kauf eines Revolvers) eine erzählerische Alternative, ein *bivium*, einen Scheideweg.

Während die Kerne chronologische und logische Funktionalität besitzen, also wichtig für die Geschichte bzw. den Spielfortschritt sind, haben Katalysen keinen alternativen Charakter, sondern lediglich eine chronologische Funktion. Sie sind gewissermaßen Parasiten, die sich an der logischen Struktur der Kerne nähren und den Raum zwischen zwei Momenten der Geschichte beschreiben. Im Beispiel könnte eine Beschreibung folgen, mit welcher Mühe sich der Spieler durch den Gully zwängt, deren Fehlen in frühen Adventures nur dem mangelnden Speicherplatz geschuldet ist, später jedoch oft eingesetzt wird, um den Unterschied zwischen zeitlichen und logischen Folgerungen zu verschleiern: "With great effort, you open the gully far enough to allow entry" statt "The gully is open". (Daß sie keine logische Bedeutung haben, zeigt sich schon daran, daß man sie über den Befehl "VERBROSE [ON, OFF]" abschalten kann.) Die Kerne sind also die Risikomomente der Erzählung, die Stellen an denen der Spieler Entscheidungen zu treffen hat, die Katalysen hingegen sind die Sicherheitszonen und Ruhepausen, die Momente, in denen das Spiel aufgrund bestimmter Entscheidungen ohne Eingriffe weitergeht.

Die Erzählung von Adventures erscheint also als System von funktionalen Öffnungen und Schließungen, gewissermaßen als teleologischer Staffellauf, in dessen Verlauf jedes Objekt an einem bestimmten Ort zu einer bestimmten Zeit durch den Spieler zu seinem vorbestimmten Ziel, zu seinem Übergabezusammenhang findet. Als Aufgabe des Spielers erscheint also auf erzählerischer Ebene das Auslösen von Katalysen, weshalb ich Adventures *entscheidungskritische* Spiele nennen möchte.

### 3. Topographie

Die Erzählungen von Adventures sind nun auf diskrete "Räume" (*rooms* ist nebenbei ein Ausdruck der Höhlenforschung) verteilt. Die Räume bilden die Orte, an denen Kerne lokalisiert sind und von denen Katalysen ihren Ausgang nehmen. Indem der Spieler Probleme löst, funktionale Schließungen vornimmt, durchläuft er notwendigerweise zugleich die Topographie der Adventurewelt. Oder umgekehrt: Wenn Anfang und Ende eines Adventures jeweils Orte sind, dann ist der Sinn des Spiels, und zugleich die einzige Möglichkeit, es 'sinnvoll' zu spielen, vom ersten Ort zum letzten Ort zu gelangen und *en passant* alle Katalysen herbeizuführen, die nötig sind, um von einem Ausgangspunkt namens Spielbeginn zu einem Endpunkt namens Spielende zu gelangen. Ich möchte drei Beispiele erwähnen, die eine vergleichbare Struktur haben.

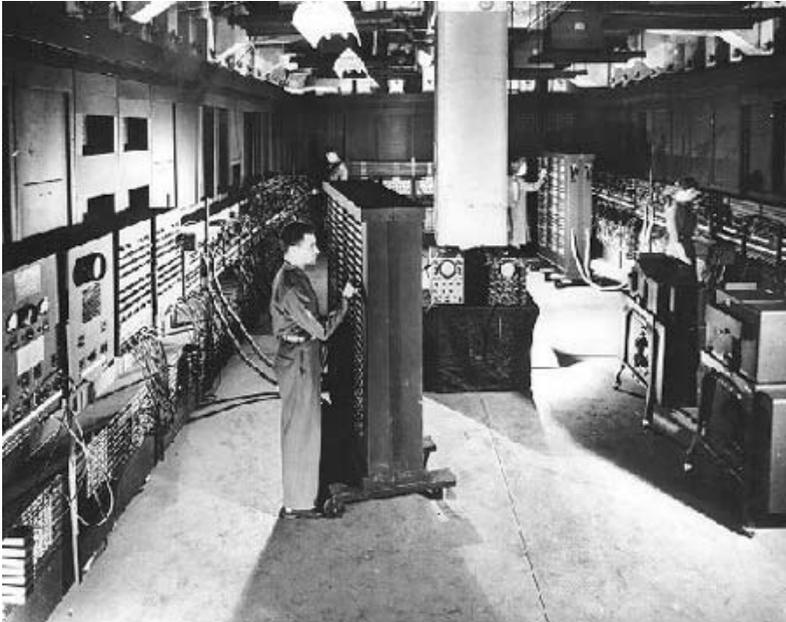


Abb.6: Die U-förmige Anlage des ENIAC-Rechners

**1. Flowcharts:** Blickt man auf die Geschichte der Programmierung hardwaregewordener Turing-Maschinen, dann zeigt sich, daß es eine Ähnlichkeit von "Entscheidungsproblem" und entscheidungskritischem Adventurespiel gibt. Von Neumanns und Goldstines "Planning and Coding"<sup>15</sup> ist (Wolfgang Hagen hat zuletzt darauf hingewiesen<sup>16</sup>) die Verteilung eines Plots in Form eines bekannten mathematischen Problems auf die Topographie einer Rechnerarchitektur. Der U-förmige *ENIAC* war bekanntlich raumgroß und die Einheiten für Addition,

Multiplikation usw. waren sichtbar im Raum verteilt. Von Neumann geht daher nicht von einem Konzept der Sprache, sondern von einem der Kartographie aus. Seine Flußdiagramme beschreiben einen Wegzusammenhang zwischen Eingang und Ausgang, auf den bestimmte Entscheidungssituationen und Übertragungen, also mit Barthes: Kerne und Katalysen, verteilt sind. (Deswegen sind Adventurespiele auch so leicht in der Neumannschen Notation anschreibbar.) Die Frage nach dem, was der Spieler ist, erscheint unter diesen Voraussetzungen als Frage nach dem, was eigentlich durch das Flußdiagramm fließt. Dazu John von Neumann:

"[I]t seems equally clear, that the basic feature of the functioning of the code in conjunction with the evolution of the process that it controls, is to be seen in the course which the control C takes through the coded sequence, paralleling the evolution of that process. We therefore propose to begin the planning of a coded sequence by laying out a schematic of the course of C through that sequence, i.e. through the required region of the Selectron memory. This schematic is the flow diagram of C."<sup>17</sup>

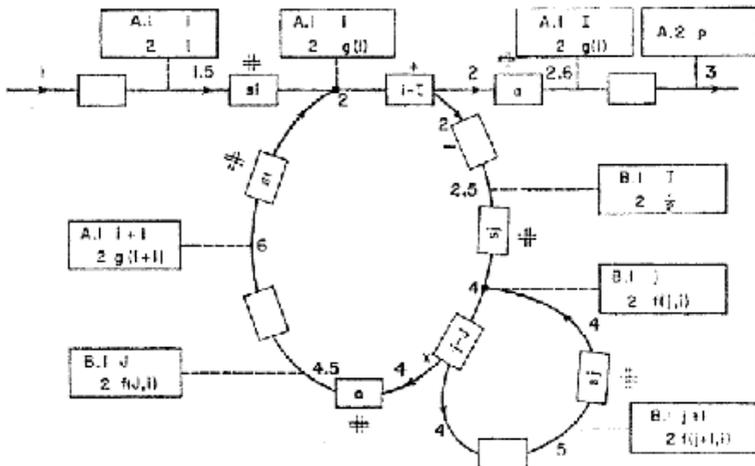


FIG. 7.9

Abb.7: Flußdiagramm nach Goldstine / von Neumann

Dabei bezeichnet C einfach die Speicherinhalte, die auf der Verfahrensrouten des Diagramms prozessiert werden. Die Gesamtheit dieser abgefragten Speicherinhalte ist das 'Objekt', der Datensatz mit seinen Attributen und Eigenschaften, der den Spieler repräsentiert. Der Avatar 'fließt' gewissermaßen durch die Präskriptionen eines Verfahrensweges wie ein Formular durch einen Dienstweg, auf dem bestimmte Eintragungen und Löschungen vorgenommen werden müssen, damit der nächste Entscheidungsort erreicht werden kann. Nun findet das Adventurespiel in der Ungewißheit über den künftigen Verfahrensweg statt, ja es besteht sogar

darin, den einzig richtigen für jede einzelne Situation zu ermitteln. Der Spieler ist also mit der Bearbeitung eines Datensatzes beschäftigt, der ausschließlich auf der taktischen Ebene jeweils so zu manipulieren ist, daß er verfahrenskonform wird. Die Alternativboxen oder "Durchgangspunkte" eines Adventures bedürfen nicht der rhetorischen Leistung eines Lebenslaufes, die darin besteht, eine Einheit durch die wiederholte "Integration von Nichtselbstverständlichkeiten" in ein Zeitschema herzustellen.<sup>18</sup> Sie kontrollieren lediglich die jeweils situative Beschaffenheit des sie durchfließenden Datensatzes. Diesen Datensatz – den Avatar, Protagonisten oder poetischer "Engelskörper" – zu manipulieren, also in taktischen Entscheidungen dessen Attribute und Eigenschaften zu verändern und ein Hindurchgleiten zu ermöglichen, ist Aufgabe des Spielers. Adventures haben, kurz gesagt, nichts mit dem viatorischen Prinzip des klassisch/romantischen Bildungsromans zu tun.

**2. Labyrinth:** Adventurespieler zeichnen beim Spielen Karten, weshalb das Spielen eine doppelte Entdeckungsbewegung, nämlich von pro-grammierter (also immer schon geschriebener) Erzählung und Karte bedeutet. Im Verlauf des Spiels werden also immer nur Teile der Topographie und der Erzählung überschaubar, mit seinem Ende kippt jedoch partikulare Ansicht in globale Übersicht, Verwirrung des Moments in Epiphanie der Ordnung. Dies ist auch das Prinzip des neuzeitlichen, des multikursalen Labyrinths, das als Figur spätestens seit Comenius *Labyrinth der Welt* Erzählung mit Topographie verschmilzt.<sup>19</sup>



Abb.8: Labyrinth von Versailles (1674) mit optimierter Route von Charles Perrault

Ich erwähne nur ein Labyrinth, nämlich jenen Irrgarten in Versailles, der 1674 nach Entwürfen des königlichen Gartenarchitekten André le Nôtre vollendet wurde. In diesem versteckt sind 39 Skulpturen nach Äsop, die der Führer Charles Perraults beschreibt. Bemerkenswert ist, daß Sébastien LeClerc's Kupferstich einen Ariadnefaden verzeichnet, der nicht nur dem bloßen Herauskommen aus dem Labyrinth dient, sondern den kürzesten (und nahezu schleifenfreien) Weg beschreibt, der an allen 39 Skulpturen entlangführt. Schreibt man nun noch das Labyrinth als Graphen an, dann zeigt sich, daß zwei Drittel aller Kerne (oder graphentheoretisch: Knoten) mit Skulpturen besetzt sind. Die Routenplanung des Perraultschen Führers erscheint also als *Programm* oder *Routing* im heutigen Sinne, indem sie (Besucher-)Ströme von einem Eingang (*Input*, Sender) über einen optimierten Entscheidungsweg zu einem Ausgang (*Output*, Empfänger) leitet. Daß dabei die Reihenfolge der Knoten oder Skulpturen wichtig sein kann, daß sie – in einer bestimmten Reihenfolge abgesprochen – zusätzlichen Sinn machen, also die 'Gestalt' einer Erzählung bekommen, ist eine zusätzliche Option.

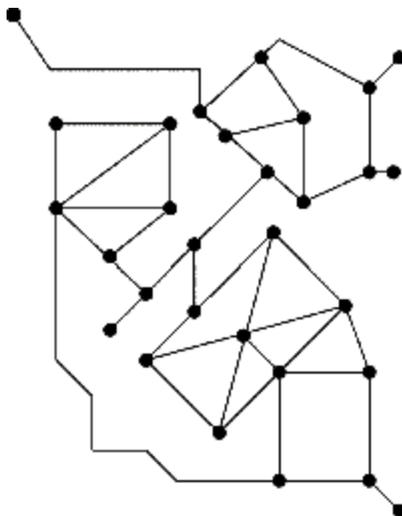


Abb.9: Graph des Labyrinths von Versailles



Abb.10: Leonhard Eulers Königsberger Brücken

Dieses touristische Problem wiederholt sich bekanntlich einige Jahrzehnte später in Königsberg, wenn Leonhard Euler einen Weg sucht, wie sich alle Brücken über den Pregel überschreiten lassen ohne eine davon zweimal ablaufen zu müssen.<sup>20</sup> Seitdem heißt ein Graph  $G$  "eulersch", wenn es einen geschlossenen Kantenzug in  $G$  gibt, der alle Kanten von  $G$  genau einmal durchläuft. Diese Vermeidung von Redundanz behandelt Sehenswürdigkeiten als Sehensnotwendigkeiten und macht aus Irrgärten mathematische Graphen.

## 4. Graphentheorie

Damit komme ich zum letzten Punkt, an dem die Fäden wieder etwas zusammenlaufen. Die Erzählungen von Adventurespielen sind graphentheoretisch "Bäume", also zusammenhängende, kreisfreie Graphen mit (so sie nicht trivial sein sollen) mindestens zwei Blättern.<sup>21</sup> Die Kreisfreiheit von Bäumen garantiert logischerweise, daß jeder Weg in einem Baum ein "Pfad", also der kürzeste Weg zwischen zwei Knoten ist. Dies birgt einige Implikationen für die Erzählung. Wenn die Größe  $|E|$  eines Graphen durch die Anzahl seiner Kanten bestimmt ist und die Erzählung eines Adventures ein Baum ist, dann ist die 'richtige' Geschichte, also die, die es spielerisch herzustellen gilt, diejenige, die  $|E|$  am nächsten kommt. Angenommen die Geschichte eines Adventures hat 12 Kanten (oder mit Barthes: Katalysen), von denen sechs zu Blättern führen, also einem Spielende, das z.B. die Form des Todes des Spielers annimmt, dann bedeutet dies, daß das 'richtig' oder erfolgreich zu Ende gespielte Spiel sechs Kanten hat und mindestens fünf Fehlentscheidungen bereithält.

Die Illustration zeigt einen solchen Baum, wobei der unterste Knoten  $s$  (*source*) der Spielbeginn und der oberste  $t$  (*target*) das erfolgreiche Spielende, der Graph also "gerichtet" ist. Gestrichelte Kanten signalisieren Fehlentscheidungen, die zum Tod des Spielers führen. Wie ersichtlich, bedeutet das gelungene Spiel das Durchlaufen einer *möglichst großen Anzahl* von Kanten, nicht jedoch *aller* Kanten. Spielen erscheint als Versuch, ein Ende möglichst lange hinauszuschieben *ohne* redundant zu werden, und zwar genau so lange, bis alle funktionalen Schließungen vollzogen sind, bis gewissermaßen kein erzählerisches Legat übrig bleibt.

Zur anderen Hälfte, die den zweiten Graphen ausmacht, sind Adventurespiele Topographien. Sie basieren auf Labyrinthen, auf die eine Geschichte pointiert, verteilt werden muß. Betrachtet man Karten von Adventures und rekonstruiert den dazugehörigen Spielweg, so wird deutlich, daß es sich nicht um Bäume handelt, sondern um zusammenhängende Graphen mit zwei ausgezeichneten Knoten, nämlich den ersten Raum  $s$  und den letzten Raum  $t$ . Folglich müssen der Baum der Erzählung und der Graph des Raumes nur in diesen beiden Punkten zur Deckung



Flußdiagramm) von  $s$  nach  $t$ , wobei – getreu dem Flußerhaltungssatz – das Gut an keinem Knoten außer  $s$  eingebracht werden und an keinem außer  $t$  austreten darf, denn sonst gäbe es ja ein Sicherheitsleck. Und damit wären wir wieder bei William Crowther und den Routing-Problemen der *ARPAnet*-Konstrukteure, denn nichts anderes beschreibt Lickliders entscheidender Entwurf von 1968 als das graphentheoretische Problem, wie über ein Netzwerk von *nodes* und Kanälen (oder Kanten) Datenpakete optimal von  $s$  nach  $t$  transportiert werden können.<sup>22</sup>

Im Netz werden diese Probleme jedoch nicht durch Spieler oder Operateure gelöst, sondern aus Komplexitätsgründen durch Softwareintelligenz, was in den 60ern zu einem Rendezvous von Algorithmen und Graphentheorie führte.<sup>23</sup> Die Tabellen, mit denen Crowther arbeitete, um nicht nur die intakten und defekten Leitungen des *ARPAnet*, sondern zugleich auch die von seiner Frau erfaßten Höhlenein- und ausgänge zu verwalten, sind einfache Adjazenzlisten, durch die man Graphen gängigerweise codiert. Und nicht zufällig entsteht um 1972 der sogenannte *DFS* (*Depth First Search*)-Algorithmus, der durch bestimmte Auswahlstrategien wie Kantenschichtverfahren kürzeste Wege (also Pfade) von  $s$  nach  $t$  durch einen Graphen sucht.<sup>24</sup> Dies ist ein ökonomisches Problem, das (wie schon in Elektrizitäts- und Telefonnetzen) als Minimierung von "Überführungskosten" gehandelt und nach dem Greedy-Prinzip organisiert wird, welches besagt, daß *lokal* beste Lösungen auch *global* die besten sein werden.

Dies könnte beispielsweise nahelegen, die "Literatur" von *Adventures* als Funktion einer graphischen Kantengewichtung zu lesen, die Übergangswahrscheinlichkeiten herstellt. Daß die Benutzung einer Banane zum Öffnen einer Tür geringe Funktionswahrscheinlichkeit hat, versteht sich (lebensweltlich-stereotyp) von selbst, daß die Überwindung eines Gegners die Erzählung mit hoher Wahrscheinlichkeit voranbringt, ebenfalls. Die Strings, die an jeder Risikosituation namens Knoten auf dem Bildschirm erscheinen, legen bei 'richtiger' Lektüre bestimmte Wahrscheinlichkeiten nahe. Man könnte vielleicht sagen, daß so etwas wie 'poetische Notwendigkeit' den Spieler bei seiner Bildung von Metonymien instruiert. Die von ihm getroffenen Entscheidungen basieren auf seiner Kompetenz, eine bestimmte Art von narrativer Plausibilität zu erkennen. Daß *Adventures* sich stark an Gattungskonventionen (Detektivgeschichte, Fantasy usw.) halten, dient wahrscheinlich dazu, diese Übergangswahrscheinlichkeiten zu modellieren, oder genauer: Unwahrscheinlichkeit zu senken. Hayden White nennt das in Anlehnung an Northrop Frye "patterns of meaning" und meint damit, "einer Ereignisfolge eine Plotstruktur zu verleihen, so daß sich ihr Charakter als verstehbarer Prozeß durch ihre Anordnung als eine Geschichte von ganz bestimmter Art [...] offenbart."<sup>25</sup> Eine 'Geschichte bestimmter Art' gewichtet also die Kanten des Graphen. In Graphen gibt es zwar kein narratives Geschehen, sondern nur Anfangs- und Enddaten, doch bieten sie als "Protokollanten der Kontingenz" (White) gerade deshalb die Möglichkeit, daß "Geschichte" auf ihnen aufsetzen kann. Eine Diskursarchäologie

des Adventurespiels bestünde also darin, nicht die Geschichten von Adventures, sondern ihre Graphen und Algorithmen zu lesen, nicht die Inhalte mit pädagogischer Sorge zu interpretieren oder philologischer Sorgfalt zu vergleichen, sondern die Möglichkeitsbedingungen der Aussagen selbst in den Blick zu bekommen.

## 5. Datenbanken erzählen

Zum Schluß möchte ich die losen historischen Fäden zu zwei Thesen zusammenziehen, in denen sich die verschiedenen Aspekte von Interaktion, Narration und Wissensverwaltung vielleicht verbinden könnten.

*Erstens* möchte ich an Vannevar Bushs prominente Phantasie einer netzwerkförmigen Wissensorganisation erinnern. Es scheint mir alles andere als zufällig, daß einerseits Adventurespiele von Leuten entwickelt wurden, die im Netzwerkrouting arbeiteten und daß andererseits Bush die Navigation durch die Datenbestände seines *MEMEX* mit dem *routing* in automatisch vermittelnden Telefonnetzen vergleicht. Die Arbeit der Lektüre besteht – so schreibt Bush – darin, Anknüpfungspunkte zu finden und diese als Verknüpfungen oder links zu realisieren: "Selection by association, rather than by indexing". Wissen entsteht, wie Bush als Koordinator von 6.000 Wissenschaftlern während des Zweiten Weltkriegs lernen konnte, darin, Zuhandenes unter bestimmten Gesichtspunkten zu verknüpfen:

"A record, if it is to be useful to science, [...] must be stored, and above all it must be consulted [...] In fact every time one combines [...] facts in accordance with established logical processes, the creative aspect of thinking is concerned only with the selection of data and the process to be employed".

Banaler gesagt, geht es in Adventurespielen nicht darum, die Datenbank der Spielwelt auszulesen (das würde Bush allenfalls "indexing" nennen), sondern darum, Verknüpfungen herzustellen: Waffen mit Gegnern, Schlüssel mit Türen, Fragen mit Antworten. Es lohnt jedoch, dabei einen kühlen oder distanten Blick zu behalten, der erkennen läßt, wie Waffen, Gegner, Schlüssel, Türen, Fragen oder Antworten allesamt *records* oder Datensätze sind, zwischen denen Verbindungen hergestellt werden.

Die Logik solcher Verknüpfungen während des Lesens ist, so Bush weiter, 'menschlich', weil sie unwahrscheinlich ist. Daher sollte das, was wahrscheinlich (und leichter berechenbar) ist, besser den Maschinen überlassen werden, das aber, was unberechenbar ist, zumindest in eine benutzerfreundliche Form gebracht werden.

"A new symbolism, probably positional, must apparently precede the reduction of mathematical transformations to machine processes. Then, on beyond the strict logic of the mathematician, lies the application of logic in everyday affairs. We may some day click off arguments on a machine with the same assurance that we now enter sales on a cash register."

Datensätze, Dokumente oder Diskurselemente haben Positionen oder Orte. Sie liegen in der Topographie eines Netzes oder einer Karte als Knoten vor, die durch Kanten verbunden werden, die der Benutzer des *MEMEX* einzeichnet. Die fortwährend wiederholte Urszene lautet: "Before him [dem Benutzer] are the two items to be joined". Was aus dieser lesenden Codierung von Dokument-Knoten durch Adjazenzlisten entsteht, ist ein Weg (*trail*) durch eine topographische Wissensordnung, die Bush wiederum 'Labyrinth' nannte. "Thus he [der Benutzer] builds a trail of his interest through the maze of materials available to him. And his trails do not fade". Die Korrespondenzen könnten deutlicher nicht sein. Ich bin in einem Raum 13 mit Item 47 und Item 73 der Datenbank, genannt "Gefängnis", "Schlüssel" und "Tür". Ich verknüpfe Item 47 mit Item 73, indem ich "benutze Schlüssel mit Tür" anklicke. Das Bit für "offen" an Item 73 wird gesetzt, die Tür ist offen. Ich bewege mich in Raum 14 weiter und ziehe eine Spur auf der Karte des Spiels, der Karte auf der die zu verknüpfenden Dinge positioniert sind. Ich speichere den Spielstand ab und meine Spur verschwindet nicht. Adventures sind Datenbanken im Sinne von Bush: Eine positionale Ordnung der Wissensobjekte, zwischen denen es *links* herzustellen gilt.

Was ist aber (und dies wäre der zweite Punkt) das besonders "menschliche" an dieser Art Datenverwaltung, das, was Bush unberechenbar oder unwahrscheinlich nennt? Wenn die Karte des Spiels eine verräumlichte Datenbank ist, dann ist es die Erzählung, die diese Datenbank verzeitlicht. Die Daten müssen durchlaufen werden, weshalb Lösungsanleitungen ja auch "walkthroughs" heißen. Erzählung ist eine bereits im Netz gelegte Spur, die ich durchlaufen muß. Umgekehrt – und damit komme ich wieder auf Hayden White zurück – ist Erzählung eine Methode, unwahrscheinliche Ereignisse wahrscheinlich zu machen. Erzählung ist also das, was eine Spur erkennen läßt und ihre Nachzeichnung während des Entstehens ununterbrochen instruiert und lenkt. Erzählung ist das, was einen Schritt, einen Klick, eine Entscheidung plausibilisiert, was an eine Form erinnert und die Kanten des Netzes gewichtet. Erzählung ist, was Übergangswahrscheinlichkeiten herstellt. Etwas wie 'poetische Notwendigkeit' unterrichtet den Spieler bei seiner Herstellung von Verknüpfungen, bei der Rekonstruktion einer Plotstruktur. Die von ihm getroffenen Entscheidungen basieren auf seiner Kompetenz, eine bestimmte Art von narrativer Plausibilität zu erkennen. Daß Adventures sich stark an Genrekonventionen halten, dient wahrscheinlich dazu, Übergangswahrscheinlichkeiten zu modellieren, oder genauer: Unwahrscheinlichkeit zu senken.

Ich mache diesen Vorschlag nicht zuletzt deshalb, weil Computerspiele ganz besondere Formen von Ereignissen produzieren. Es handelt sich um Ereignisse, die eine Virtualität und eine Aktualität haben. Virtuell deshalb, weil sie in Datenbank und Fabel immer schon geschrieben stehen und sich in gewissem Sinne immer schon ereignet haben. Aktuell deshalb, weil sie in jedem neuen Spielen aus der Möglichkeit in die Wirklichkeit geführt werden und aus dem Kalkül in die Zeit übertreten. Es sind gewissermaßen versicherungstechnische Ereignisse, wie Autounfälle, die mit dem Unterschreiben des Vertrages und dem Zahlen der Beiträge immer schon virtuell passiert sind, bevor sie sich aktuell ereignen. Sie sind zugleich gespaltene Ereignisse, Ereignisse mit zwei Hälften, einer sichtbaren und einer unsichtbaren. Sie entstehen aus einem Zusammenspiel von Menschen mit Maschinen, Eingaben und Programmen, kurzum aus einer Interaktion, die eine kommensurable und eine inkommensurable Seite hat.

Diese Seiten schreiben verschiedene Formen der Geschichte: eine Geschichte in einem anderen Medium und eine Geschichte des Mediums "in its own terms", wie Timothy Lenoir sagen würde. Das ist einfach nachzuprüfen: Man braucht nur einen Spieler zu fragen, was er gerade getan hat, und er wird eine Geschichte der Art erzählen: "Erst habe ich mir eine hölzernen Hand besorgt, dann bin ich damit zum Termitenzirkus gegangen, habe die Termiten auf eine Jagdtrophäe des Bösen abgesetzt, der hat dann vor Wut seinen Stoch durchgebrochen, usw."<sup>26</sup> Ich frage mich, wie die andere Geschichte zu schreiben wäre, die Geschichte des Mediums in seinen eigenen Begriffen. Unter Unix gibt es ja den schönen Befehl *HISTORY*, der eine Chronik der letzten Eingaben am Terminal zurückgibt. In diesem Sinne müßte man wohl die Befehle, die gesetzten Flags und Attribute entlang eines Timecode verzeichnen, die sich ereignen, wenn eine hölzerne Hand ergriffen wird, die sich ereignen, wenn das Spielerobjekt dem Termitenzirkusobjekt nahekommt, die sich ereignen, wenn Hand und Termiten sich zusammentun. Eine solche Geschichte bestünde wohl aus "Protokollen" (oder eben "Protokollanten der Kontingenz") – der Kontingenz deshalb, weil menschliche Eingaben unwahrscheinlich sind. Jedenfalls wäre es wohl keine Erzählung.

Vielleicht ist die Frage nach Ereignis und Geschichte diejenige, an der sich die beiden Möglichkeiten treffen, Computerspiele anzusehen: Der einen, die sich gewissermaßen die menschliche Seite anschaut, und der anderen, die sich die maschinische Seite anschaut. Eine Frage, die vermittelt zwischen den Begriffen Interaktion, Narration und Wissensorganisation und zugleich zwischen technikzentrierten und performativen Fragen. In diesem Sinne könnte man vorschlagen, Geschichte als Interface zu begreifen. Erzählung ist das, was Programmereignisse ermöglicht ohne daß Spieler/User dabei zählen müssen. Denn ein Adventurespiel auf Datenbankebene zu spielen ist ziemlich unmöglich, wenn keine Erzählung da ist. Es wäre nicht nur müßig, sondern würde auch unser Vorstellungsvermögen übersteigen und keinen Spaß machen, mit einem

Kommando die Datensätze 47 und 73 zu verlinken. Es ist aber sehr leicht, den Befehl "benutze Schlüssel mit Tür" zu tippen oder zu klicken.

Adventurespiele sind zugleich etwas anderes als das Netz, weil sie erstens endlich sind und weil es zweitens einen vorgeschriebenen Weg durch ihre Datenbestände zu nehmen gilt. Für genau dies, für die Rekonstruktion und den Nachvollzug einer gelegten Spur wäre Erzählung dann das Interface. Sie wäre das, was eine Archivarbeit ermöglicht, durch die tatsächlich herauskommt, wie es "wirklich gewesen ist" und immer sein wird. Und das sollte eine Medienwissenschaft zumindest anregen, die Erzählung von Computerspielen nicht bloß vom Erzählten oder der Konkurrenz zu Filmen und Büchern her zu begreifen, sondern von den Möglichkeiten des computergestützten Erzählens her, von dem, was überhaupt mit Datensätzen, Verknüpfungen und Karten erzählbar ist und was nicht.

## Fußnoten

---

1. Leicht veränderte Fassung meines Aufsatzes: "Es mag wohl labor intus sein". Adventures Erzählen Graphen", in: Ulrike Bergermann / Hartmut Winkler (Hrsg.): *TV-Trash. The TV-Show I Love to Hate*, Marburg 2000, S. 85-106.
2. Katie Hafner / Matthew Lyon, *Where Wizards Stay Up Late. The Origins of the Internet*, New York 1996; William Crowther, Frank Heart, Robert Kahn, Severo Ornstein und David Walden, "The Interface Message Processor of the ARPA Computer Network", Paper für die *Spring Joint Computer Conference of the American Federation of Information Processing Societies*, 1970.
3. Zum Folgenden ausführlich: Alexander Clark Bullitt, *Rambles in Mammoth Cave During the Year 1844 by a Visiter(!)*, New York (Reprint) 1973; Roger W. Brucker / Richard A. Watson, *The Longest Cave*, New York 1976; Duane De Paibe, *Gunpowder from Mammoth Cave. The Saga of Saltpetre Mining Before and During the War of 1812*, Hays, Kansas 1985; Joy Medley Lyons, *Mammoth Cave. The Story Behind the Scenery*, Las Vegas 1991; [www.mammothcave.com](http://www.mammothcave.com); [www.nps.gov/macac/macahome.htm](http://www.nps.gov/macac/macahome.htm); [www.nps.gov/macac/slh.pdf](http://www.nps.gov/macac/slh.pdf).
4. Dazu gehörte beispielsweise auch ein Programm, das die über Tastatur eingegebenen Notationsdaten als Karte auf einem Plotter ausgeben konnte (beispielsweise – in Vergessenheit des antiken bivium – "Y2" für einen Scheideweg mit Haupt- und Nebenausgang). Zum folgenden vgl.: *A History of 'Adventure'* ([people.delphi.com/rickadams/adventure/a\\_history.html](http://people.delphi.com/rickadams/adventure/a_history.html)); Tracy Kidder, *Die Seele einer neuen Maschine. Vom Entstehen eines Computers*, Reinbek 1984,

- S. 88-93; Steven Levy, *Hackers. Heroes of the Computer Revolution*, London 1984, S. 281-302; Hafner / Lyon, a.a.O., S. 205ff.
5. Mitteilung von Melburn R. Park vom Department für Neurobiologie der Universität Tennessee.
  6. Jacques Derrida: "Point de la folie – maintenant l'architecture", in: *Psyché. Invention de l'autre*, Paris 1987.
  7. Dazu Hafner / Lyon, a.a.O., S. 83-136.
  8. Wolfgang Hagen, "Der Stil der Sourcen" ([www.is-bremen.de/~hagen](http://www.is-bremen.de/~hagen)).
  9. Roman Ingarden, *Vom Erkennen des literarischen Kunstwerks*, Tübingen 1968.
  10. Zu einer wittgenstein'schen Lektüre dessen, 'was der Fall ist' vgl. Heinz Herbert Mann, "Text-Adventures. Ein Aspekt literarischer Softmoderne", in: *Besichtigung der Moderne. Bildende Kunst, Architektur, Musik, Literatur, Religion. Aspekte und Perspektiven*, hg. H. Holländer / C.W. Thomsen, Köln 1987.
  11. Immanuel Kant, *Kritik der reinen Vernunft*, Hg. R. Schmidt, 3. Aufl. Hamburg 1990, S. 115-125.
  12. Martin Heidegger, "Sprache", in: *Überlieferte Sprache und technische Sprache*, St. Gallen 1989, S. 20-29.
  13. Alain Robbe-Grillet, *Der Augenzeuge*, Frankfurt a.M. 1986, S. 159.
  14. Roland Barthes, "Einführung in die strukturelle Analyse von Erzählungen", in: *Das semiologische Abenteuer*, Frankfurt a.M. 1988, S. 102-143. Das Beispiel ist willkürlich; zur Funktionalität der Ereignislogik könnte bspw. auch Claude Bremond (*Logique du récit*, Paris 1973) herangezogen werden ("raconter, c'est énoncer un devenir", S. 325); oder Arthur C. Danto (*Analytical Philosophy of History*, Cambridge 1965), dem Erzählen als logischer Dreischritt erscheint: "1. X is F at t-1; 2. H happens to X at t-2; 3. X is G at t-3" (S. 236); oder Wolf-Dieter Stempels Minimalforderungen des Erzählens wie "resultative Beziehung" zwischen Ereignisgliedern, "Referenzidentität des Subjekts", "Solidarität der Fakten" usw. ("Erzählung, Beschreibung und der historische Diskurs", in: *Geschichte -Ereignis und Erzählung*, Hg. R. Koselleck / W.-D. Stempel, München 1973, S. 325-346 [Poetik und Hermeneutik V]).
  15. Zum Folgenden vgl.: Herman H. Goldstine / John von Neumann, "Planning and Coding Problems for an Electronic Computing Instrument", in: John von Neumann, *Collected Works*, Hg. A.H. Taub, Bd. 5, New York 1963, S. 81-235; Herman Goldstine, *The Computer from Pascal to von Neumann*, 2. Aufl. Princeton 1993; Presper Eckert Interview ([www.si.edu/resource/tours/comphist/ekkert.htm](http://www.si.edu/resource/tours/comphist/ekkert.htm)); ENIAC History Overview ([seas.upenn.edu/~museum/hist-overview.html](http://seas.upenn.edu/~museum/hist-overview.html)); Arthur W. Burks / Alice R. Burks, "The ENIAC. First General-Purpose Electronic

- Computer", in: *Annals of the History of Computing*, 3,4/1981, S. 310-389; Arthur W. Burks, "From ENIAC to the Stored-Program Computer. Two Revolutions in Computers", in: *A History of Computing in the Twentieth Century*, Hg. N. Metropolis / J. Howett / G.-C. Rota, New York / London 1980, S. 311-344.
16. Wolfgang Hagen, *Von NoSource zu Fortran*, Vortrag auf dem Kongress "Wizards of OS", Berlin 16.07.99 ([www.is-bremen.de/~hagen/notofort/NoSourceFortran/index.htm](http://www.is-bremen.de/~hagen/notofort/NoSourceFortran/index.htm)).
17. Goldstine / von Neumann, a.a.O., S. 84.
18. Niklas Luhmann, "Erziehung als Formung des Lebenslaufs", in: *Bildung und Weiterbildung im Erziehungssystem. Lebenslauf und Humanontogenese als Medium und Form*, Hg. D. Lenzen / N. Luhmann, Frankfurt am Main 1997, S. 11-29, S. 18.
19. Max Bense, "Über Labyrinth", in: *Ästhetik und Engagement*, Köln / Berlin 1970, S. 139-142; Helmut Birkhan, "Laborintus – labor intus. Zum Symbolwert des Labyrinths im Mittelalter", in: *Festschrift für Richard Pittioni*, Wien 1976, S. 423-454; Gaetano Cipolla, *Labyrinth. Studies on an archetype*, New York 1987; Penelope Reed Doob, *The Idea of the Labyrinth from Classical Antiquity through the Middle Ages*, Ithaca / London 1992; Umberto Eco, "Kritik des Porphyrischen Baumes", in: *Im Labyrinth der Vernunft. Texte über Kunst und Zeichen*, Leipzig 1990, S. 89-112; Abraham Moles / Elisabeth Rohmer / P. Friedrich, *Of Mazes and Men – Psychology of Labyrinths*, Strasbourg 1977; Karl Kerényi, "Labyrinth-Studien", in: *Humanistische Seelenforschung*, Wiesbaden 1978, S. 226-273; Hermann Kern, *Labyrinth. Erscheinungsformen und Deutungen*, München 3. Aufl. 1995; Wolfgang Haubrichs, "Error inextricabilis. Form und Funktion der Labyrinthabbildung in mittelalterlichen Handschriften", in: *Text und Bild. Aspekte des Zusammenwirkens zweier Künste in Mittelalter und früher Neuzeit*, hg. C. Meier / U. Ruberg, Wiesbaden 1980, S. 63-174.
20. Leonhard Euler, *Solutio problematis ad geometriam situs pertinentis*, Comment. Acad. Sci. I. Petropolitanae 8/1736, S. 128-140.
21. Einführend Reinhard Diestel, *Graphentheorie*, Berlin 1996.
22. J.C.R. Licklider, *The Computer as a Communication Device*, in: Science and Technology, April 1968 (Reprint bei DEC, Systems Research Center, Palo Alto 1990), S. 32ff. Eine vergleichbare packet-Struktur brachte ein 1960 projektiertes und pünktlich zum Vietnamkrieg implementiertes Netzwerk hervor, nämlich in Gestalt des Containers als Datenpaket genormter Größe, mit Absender und Empfängeradresse, wobei das 'Gut' auf verschiedene Container verteilt wurde, die unterschiedliche Passagen nahmen (dazu David F. Noble, "Command Performance. A Perspective on the Social and Economic Consequences of Military

- Enterprises", in: *Military Enterprise and Technological Change*, Hg. M.R. Smith,, Cambridge, Mass. 1987, S. 338ff.).
23. Dieter Jungnickel, *Graphen, Netzwerke und Algorithmen*, Heidelberg 1987; Th. Emden-Weinert et al., *Einführung in Graphen und Algorithmen*, Berlin 1996 ([www.informatik.hu-berlin.de/~weinert/graphs.html](http://www.informatik.hu-berlin.de/~weinert/graphs.html)).
24. R.E. Tarjan, "Depth first search and linear graph algorithms", in: *SIAM J. Comput.* 1/1972, S. 146-160; vgl. C.Y. Lee, "An algorithm for path connection and its applications", in: *IRE Trans. Electr. Comput.* EC-10, 1961, S. 346-365; E.F. Moore, "The shortest path through a maze", in: *Proc. Internat. Symp. Theory Switching, Part II*, Cambridge, Mass. 1959, S. 285-292.
25. Hayden White, *Auch Klio dichtet oder Die Fiktion des Faktischen. Studien zur Tropologie des historischen Diskurses*, Stuttgart 1991, S. 75.
26. Ein Beispiel aus "Flucht von Monkey Island" (Lucasarts 2001).