Card Shark and Thespis: exotic tools for hypertext narrative

Mark Bernstein

Eastgate Systems, Inc.
134 Main Street
Watertown MA 02472 USA
bernstein@eastgate.com

ABSTRACT

Card Shark and Thespis are two newly-implemented hypertext systems for creating hypertext narrative. Both systems depart dramatically from the tools currently popular for writing hypertext fiction, and these departures may help distinguish between the intrinsic nature of hypertext and the tendencies of particular software tools and formalisms. The implementation of these systems raises interesting questions about assumptions underlying recent discussion of immersive, interactive fictions, and suggests new opportunities for hypertext research.

KEYWORDS: hypertext systems, fiction, narrative, Storyspace

EXOTIC NARRATIVE TOOLS

The seeming conflict between the apparent linearity of temporal experience and the nonlinearity of hypertext has provoked much theoretical discussion, especially among those with scant experience of hypertext reading (e.g. [8], [25]). Evidence that the conflict is not inherently irreconcilable is provided by the continued popularity of early hypertext fictions (e.g. [15]), the appearance of new fictions (e.g. [10]), and the flourishing secondary literature on the subject (including Nelson award-winning papers [36] and [35] and several recent books [11], [17], [19], [1], [26]). The importance of narrative in the craft of hypertext writing has been recognized from the beginning [9], for narrative is central not only to works of imagination but also to technical [3] and scholarly writing [18]. We cannot dispatch the problems of narrative to the province of artists and literary critics, for narrative questions — relating cause and effect, sequence and simultaneity, dependence and explanation — eventually challenge all writers.

HT'01 8/01 Aarhus, Denmark
® 2001 ACM ISBN 1-59113-420-7/01/0008…$5.00

I (and others) have long argued that the nature of hypertext is best discerned by studying actual hypertexts. Over the past decade, three systems — HyperCard, Storyspace, and Web browsers — have been the most common choices of hypertext fiction writers.[1]. Different writers use the same system in drastically different ways[2], but systems inevitably shape hypertexts. Are the properties of hypertext fiction, such as those observed in [6] and deplored by [25], intrinsic to hypertext, or do they arise from the idiosyncrasies of specific systems? Do the patterns we observe in hypertext fiction arise directly from hypertextuality, or through the mediation of the system?

This paper explores two exotic hypertext systems, tools suitable for hypertext narrative but dramatically unlike the tools currently in use. My motivation for describing these tools is also unusual. The customary reason for building a new system is to build a *better* system; here, I wanted to build a *strange* system, a hypertext environment that might let us step back from Storyspace and the Web in order to gain a better perspective. I do not wish to argue that these systems are better than, say, Storyspace: Storyspace is simpler, more elegant, more flexible, more widely available. I do not suggest that hypertexts written with the new system will be better than those written with other tools.

For our purposes, we need not be better, we need only be different.

In the remainder of this paper, we'll first explore a language or notation, *Card Shark,* that describes *sculptural hypertexts.* Shark is small, simple, and appears not to be very expressive, but it can readily describe complex

1 The dominance of these particular systems among fiction writers need not be ascribed to any inherent virtue or suitability to the task. Accessibility plays a crucial role, as do the accidents of history. If Guide, Trellis, or NoteCards had survived to develop a literary following, our current impressions of the nature of hypertext narrative might be quite different.

2 Compare, for example, *afternoon* [15], *Lust* [2], *The In Memoriam Web* [20], and *True North* [34]. All were written with Storyspace, but their use of links varies tremendously.

hypertext structures. Next, we embed Shark in a dramatic context; we create a simple theatrical environment that represents characters moving through space, a space through which the reader moves to witness and perhaps to participate in the action. The nature of this participation, though it shares some surface properties of interactive fiction, may escape internal contradictions that confront conventional immersive fictions.. Finally, we conclude with some thoughts on how this approach might be evaluated.

SCULPTURAL HYPERTEXT AND CARD SHARK

Conventional hypertexts take a set of unconnected nodes (or pages, or lexia) and link them together. Card Shark[3] begins with a set of nodes, all of which are connected to each other, and builds structure by removing unwanted connections. We call Card Shark *sculptural* because we create structure by removing unwanted connections, much as a sculptor may create objects by removing unwanted material. Traditional hypertext tools in this sense are *calligraphic;* we create structure by adding lines, one after another, until we have added exactly the necessary degree of connection.

Where this strategy has been employed in the past — most notably in Malloy's *its name was Penelope* and in Malloy and Marshall's *Forward Anywhere*— it has been chosen in part to deemphasize temporal sequence and textual structure [13]. Card Shark, as we shall see, foregrounds sequence and emphasizes structure.

A Card Shark node (or card) contains some text, typically a brief, focused passage. Each card may also specify constraints on the context in which it may appear. For example, `AFTER 10` requires that the node may be visited only after ten other nodes have been seen. A node that appears `BEFORE 25` may only be visited early in the reading; if it is not seen early, it will not be seen at all. A variety of constraints may be applied to a node; as in a conventional hypertext, it is likely that some nodes may never appear in any given reading.

Each card may also specify modifications it makes to the reading context, chiefly by posting assertions on a blackboard. A passage that serves to introduce a new character, for example, could `ASSERT WENDY`. Other cards that `REQUIRE WENDY` can be visited only after this introduction. Later, a passage may remove `WENDY` from the scene and `RETRACT WENDY`.

Given a collection of cards, we read them by following a simple set of rules[4]:

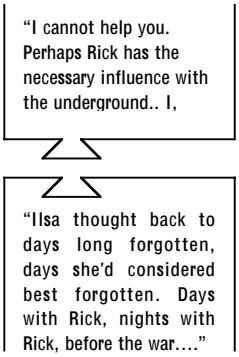

Figure 1. Card Shark nodes establish conditions that subsequent nodes must satisfy

1) The collection of cards is shuffled and the blackboard is wiped clean.
2) The reader receives 7 cards from the deck.
3) The constraints for each of the player's cards are evaluated. Cards whose conditions are not satisfied are disabled; the reader sees at most a brief title and an indication of what conditions need to be satisfied for the card to be seen.
4) The player chooses among the cards whose constraints are met, selecting a node to visit next.
5) The selected node is visited. Its full text appears (or is performed) on the screen. If the node makes assertions or modifies the environment, those actions are performed. The card remains "on the table"; we may look at it again whenever we like, but it will never be "played" again.
6) The player receives a new card, and repeats until the reading is over.

If the granularity of the lexia — the size of the "card" — is large, Card Shark's constraints describe the episodic architecture of the narrative. If the granularity is very small — individual words or phrases — the constraints describe a text generation engine. If the lexia were lines of iambic pentameter, the constraints could describe a rhyme scheme.

We can easily envision other variations. In particular, we might use one deck but maintain two separate blackboards (perhaps called Plot and Subplot). The two blackboards provide separate contexts in which cards could appear, offering the reader and the writer greater flexibility.

In conventional hypertext tools, connecting nodes in a sequence is easy but connecting nodes in a dense tangle usually requires effort. Card Shark inverts the situation; making a tangle is easy, but crafting a strict sequence requires effort. The tangle, not the link, is a Card Shark primitive. (Some versions of Storyspace include power tools for making lots of links at once, and Shark may benefit from

[3] At the first hypertext conference, Frank Halasz drew a distinction between "Card Sharks", the proponents of granular hypertexts (HyperCard, Storyspace, KMS), and "Holy Scrollers." advocates of stretchtext (Guide, Zig-Zag.).

[4] The details of these performance rules are often arbitrary and the reader may easily envision alternatives. Tinkering with rules has been a fruitful source of amusement and time-wasting during development. My current speculation is that these procedural details are not terribly important; they affect pacing and rhythm during performance, but effective writing is far more important to the performance than clever rules.

power tools for making lots of assertions at once)

Card Shark is close in spirit to van Parunak's HyperSet [28], but its goals are quite different[5]. If, we arrange the cards in a prescribed sequence before each reading instead of shuffling them, Card Shark appears to be formally equivalent to Trellis [33]. Implementing a Shark performance engine is trivial; a prototype was implemented in Macromedia Flash.

LIVENESS, CARD SHARK, AND WRITING EXERCISES

Like users of its Petri-net ancestor Trellis, Card Shark writers are naturally concerned with *liveness.* Consider a reader in the midst of reading a Card Shark hypertext. We can readily envision that a time may come when the reader examines her seven options and finds that none of the preconditions are met. The position is dead; the reader is stuck[6]. Sooner or later, this is inevitable; we will run out of cards, the story must eventually end. But the story should have a chance to play out first; writers need to take care to let the story begin before it comes to an end, to avoid stranding the reader at the outset.

Imagine, for example, a Card Shark hypertext that describes an twilight encounter in the garden on Tuesday night and its dénouement in the nearby bedroom the following morning[7]. Some actions `require night`; others `require morning`. If it is currently night and our available actions require that it be morning, we need an transitional action that moves from night to morning. Conversely, if we have been reading about events in the bedroom and there is more to learn about last night's encounter in the garden, we require a transition that moves from the morning to the previous night, from the bedroom to the garden. Indeed, if readers are not to constantly encounter dead positions, we need to provide a rich assortment of transitions[8] to facilitate movement, to shift between times, to get characters on and off stage.

These transitional nodes closely resemble a familiar class of writing exercises (see, for example, [12]). It is important to observe that the text of a node need not merely describe the state change. The text may, for example, proceed from the consequence of the asserted change without describing the transition. Indeed, some transitions may not need to be expressed, either because the reader will understand them or because we want to startle the unwary.

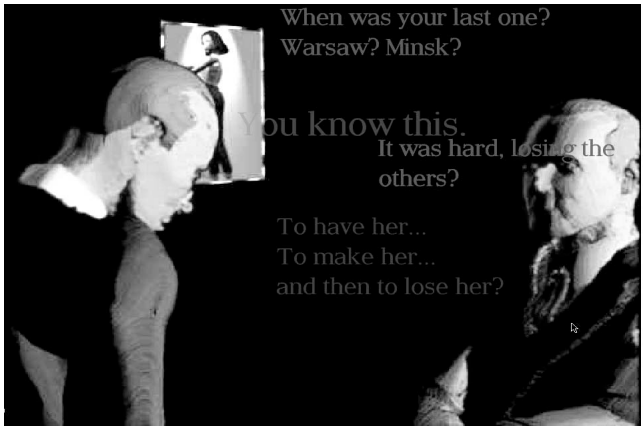

Figure 2 When text and image share the visual frame, the competition for space and for attention creates enormous problems (Scott McCloud, *Understanding Comics*). Dynamic typography can provide more space for textual expression, although the conflict between textual meaning and letterforms remains problematic. The output of the current Thespis prototype is a script— specification for performance that may be reenacted mentally by the reader or literally by a visualization program. Even if the visualization is complex, however, it need not be literal or cinematic.

Transitional nodes, when used naturally, tend also to maintain coherence and causality. At the same time, cycles do not naturally occur in Card Shark; to permit recurrence, the same passage must occur on two or more different cards. Incoherence, oscillation, repetition, and cycles are sometimes seen as inherent to hypertext[9] but may inhere more closely to the tools we happen to be using and the inclinations of their creators.

SOCIAL SHARK

Reading is often considered a solitary activity, but we might also enjoy Card Shark hypertexts with company. Extending Card Shark for collaborative reading creates *Social Shark,* and with it some interesting opportunities.

Consider two readers, Mr. Green and Ms. Blue, who meet (perhaps over the net) to read a Social Shark hypertext together. The computer unwraps a fresh deck, shuffles the cards, and deals seven cards apiece to Green and to Blue. The two readers take turns, following the rules of Card Shark; the reading continues until neither reader can continue.

Each card, in addition to its text, its preconditions, and its

[5] HypertSet was intended as an aid to taxonomic reasoning, an analytical tool for scholars rather than a performance medium.

[6] The situation is directly analogous to reaching a conventional hypertext node with no outbound links, or to an short, inescapable cycle that signals closure [6]. Early hypertext systems worried about deadness almost as much as they worried about disorientation [4]; today, we press the Back button and wonder what the fuss was about.

[7] Beginners often assume that narratives must be related in chronological sequence, that other arrangements are unnatural or "artistic". It often proves desirable to depart from chronology, both for clarity and for pacing. Events in the garden may be interesting only for what follows; unless we know the consequences we may not understand or care about the antecedents.

[8] Multivalence is not a vice [7], and these transitions need not limit themselves to their immediate business. Finding ways to add incident or illuminate character in the course of the transition is an interesting and rewarding chore.

[9] Especially by those who don't like the hypertexts they've read [8] [25].

assertions, is labeled with a green number and a blue number. Whenever a card is played, Mr. Green receives number of points specified in green, and Blue receives the number of points specified in blue. The reading continues until neither player can continue, and the winner at the end of the reading is the player with the highest score. (Extensions to additional participants are easily envisioned by adding additional score numbers. Alternatively, a third player might seek to maximize the combined blue and green score, a fourth player might seek to minimize it, and so on. Indeed, the game need not be competitive: Green and Blue might conspire toward a common goal, perhaps in a common struggle against the inherent tendency of the fictive universe.)

How might hypertext writers assign these values to cards? One simple approach identifies each player with a goal: Green is rewarded whenever Love grows between two characters, and Blue is rewarded whenever complications or misunderstandings separate them. The goal may extend to plot and subplot: Green is rewarded whenever *anyone* falls in love. But the goals of Green and Blue need not be orthogonal: Green might be rewarded when Love advances while Blue is rewarded by anarchy and chaos[10]. In this case, Green's triumph is a romantic comedy — *Pretty Woman* or *The Tempest* — while Blue's triumph is hilarity — *Horsefeathers* or *The Importance of Being Earnest*

THESPIS

Thespis[11] is a new hypertext system that extends the core idea of Card Shark by allowing many agents to participate in a single hypertext story[12]. Each agent or *actor* receives cards that describe possible actions, and each in turn selects an action to take. One agent represents reader; the reader chooses actions as she wishes. The other agents are computational structures; they choose for themselves.

Each actor has a name and a simple internal state. Each actor also has its unique function `Happiness(state)`.[13]

Actors choose actions that are likely to improve their happiness. One character may value money more than another; one may crave excitement while another avoids it. Faced with the same options, different character may choose different actions. These crude behaviors are not meant to model psychology, but merely to provide the appearance of intentionality and individuality. We are not making people; we're making theater [21].

Each actor moves across the bounded, two-dimensional space that represents the *stage*. The reader sees and hears things that happen nearby; more distant actions may be unnoticed. This spatial component neatly reifies the hypertextuality of Thespian space; rather than following *this* link and not *that* one, we are standing *here*, not elsewhere. Perhaps we are sitting and talking with Hugh, Cathy, and Kaj. Across the room, we might see Randy and Stuart arguing with Susana, but if we want to hear them we'll have to walk over there -- and then we'll miss the action now unfolding before us.

We may constrain actions in Thespis by reference to the environment and to the context. Consider a node that reads,

> *"It's getting dark. Winter is coming. I tried to remember winter – the last winter before the war. It seems so long ago."*

Without constraint, any actor might say this. But we can easily impose constraints, choosing who may say this, or to whom it may be said. We could specify when and where it can be spoken. A variety of partial constraints are provided; for example, an action that is `OnlyExcited` can only be performed if the actor is unusually agitated, and a `Private` action can only be performed if the participants don't know they're being observed.

Actors and actions are simple. Rather than create complex actors, we create simple automata that say interesting things about important matters. Actors can move (to a landmark or to another actor), they can use props (eating, for example, if they feel like it), and they can talk. The point of this computational mechanism is merely to keep the actor-automaton from breaking the theatrical illusion. We enforce a naive physics of the stage, decreeing, for example, that actors should never walk through walls. Simple logic can give rise to complex emergent behavior [32], and this aggregate behavior can be convincingly organic.

Perhaps more important, though, is the recruitment of the reader as a dramatic co-conspirator. If simple automata are well written, if they are engaging and convincing, readers will want to attribute agency, intentionality, and emotional depth to them [30].

A THESPIAN EXAMPLE

What would it be like to read a hypertext written for Thespis? Let's imagine one[14]. We'll call it *The Trojan Kids*,

---

[10] If collaborators aren't conveniently available, we might arrange for the computer to play a hand — or even several hands.

[11] Thespis, a poet of sixth century Greece, invented theater by elaborating a traditional musical form for soloist and chorus. Thespis introduced a second soloist, a character who could interact with the main character and with the chorus. Two characters introduce the possibility of dramatic conflict as well as crowded dressing rooms and arguments over top billing; liturgy becomes drama.

[12] I tried to get a second character onto the hypertext stage in "Hypertext With Characters" [5]. The earlier work seems to have been completely ignored, perhaps because its implementation was thought to be infeasible. The dearth of new hypertext systems in the late 90's is conspicuous, and may have led us to overestimate the difficulty of implementation.

[13] In the prototype, state variables included Cash, Status, Excitement, and Well-Being,. Happiness was represented as a linear combination of state variables.

[14] I beg the reader's indulgence for this lengthy exposition of the plot of an unimportant, incomplete prototype. Demonstrating narrative is a vexing problem; the only way

an experimental adaptation of Euripides' *Trojan Women*, in modern dress. It's set in a large, open, metal-roofed shed, a community center for a small village that has recently fallen to the conquering invaders. It's the night of the big school dance; life goes on. It could be France in 1940, it could be a village in Rwanda or Kosovo or Chechnya.

The reader is ALICE, uncertain, unsure, unimportant[15].

She is met at the doorway by EMILY, a plain and unpopular student who has done most of the work of arranging the dance, setting out the refreshments, getting permits from the Provisional Government. She greets us warmly,

> *Emily: Come in, come in. I'm so glad you're here. Everybody's here. Come in, let's all be together, together again. Let's celebrate the blessings of peace.*
>
> *Alice: But, Emily, we lost! After ten years, our gates lie in ruins, Greek soldiers patrol the streets, smoke rises from the palace. What blessing is this?*
>
> *Emily: Defeat is bitter, sure. But now we have peace! At last! With honor! And in our time. Now we can have our party. It's our tradition, and the Greeks gave us a permit. Go on in. Try the shrimp -- I hear the dip is really spicy!*

Emily is a Pollyana, an accommodator, a collaborationist in embryo, and the stink of a dark future hangs over her irritating cheeriness. Can she be saved?

Inside, there's quite a crowd. They're kids; most of them have simple motivations. Some hope to get drunk. Some hope for a memorable moment of basketball or Nintendo. Some hope to get lucky -- out back there's an area of secluded dunes. Some of these kids were conceived out there, just five or six years before the War began.

Others have more to say. CASSIE (Cassandra) is dark, sexy, strange. She knows stuff. She's seen Emily, for example, a few years from now, her head shaved, hounded through the street. Cassie knows that some of her friends here tonight will be in the cheering crowd. She knows that others won't make it that far. Cassie isn't popular, nobody pays attention to her stories, but she's hard to ignore; she draws boys like a flame and those boys draw girls. She's rarely alone.

POLLY (Polixena) is the old king's niece. She shouldn't be here, she doesn't know this crowd, she goes to private school. She was away from the palace when the soldiers came. She's on the run, she's escaped the patrols so far, and she's running out of options. Perhaps, if she can blend in, nobody will notice that there's a member of the royal family still at large. Perhaps she can stay free, perhaps she can live, a little longer.

FRANK and BILL are drinking from a hip flask and debating the relevance of class struggle to the War. They've been having this debate since 7th grade. Bill has just realized that he is in love with Polly, that her radiant smile makes the bare 60-watt bulbs burn more bright. He thinks he's never seen her before, that she's a new kid; he's seen her on TV a thousand times, but not in jeans and a T shirt. Frank has known for years that he's in love with Bill, and he sees this immediately and knows that it cannot come to good.

We always begin at the entrance, with Emily, but after that our experience depends on our choices. Perhaps Cassie and her coterie are hovering over the refreshments; we might join them. Cassie has plenty to say (and she can say it, because she's a prophet and prophets aren't bound by temporal constraints). Or we might first wander over to Polly. Perhaps we stand off a little ways and eavesdrop as Bill tries to chat up Polly while visions of sand dunes dance in his head. Perhaps Emily rushes up with cups of punch, urging Polly to cheer up and have a *great* time and get out and dance!

Each of the characters have things to say. They're kids; they'll tell anyone who'll listen their insights and philosophies. They move in clusters (as kids at a party do), and sometimes individuals or couples will spin off or two groups will coalesce. Topics of conversation are introduced, old topics are exhausted or discreetly abandoned. Questions and conflicts abound -- each individual and unique, but each also connected to the others and to us.

PATTERNS IN THESPIS

Is this a hypertext? *The Trojan Kids* has no links, no blue text, no map view. But these are mere externals. Thespis discloses a chunk of text and then offers the reader a set of choices, and the choice selected determines what is seen next. This sounds like a hypertext. In Thespis, a reader has seven possible choices at any moment, some of which might be unavailable.

---

to understand a work is to experience it, and even then the illustration may founder on accidents of taste, interest, or understanding. The intent of this section is to establish the example in sufficient detail to permit the reader to construct a similar hypertext in her own laboratory. The alternative—presenting a formalization of Thespis — seems futile; the history of hypertext-as-a-tuple is not bright, and neither Shark nor Thespis lend themselves to elegant formalization. Consider this to constitute the experimental section.

[15] The unimportance of the point-of-view character may be essential to Thespian hypertext. If the reader's point of view is a hero protagonist, for example, the reader is led naturally to test the limits of the possible, That's what heroes do. The drama rapidly devolves into a negotiation between the reader and the world model; the reader asks to do unexpected, the system typically responds with incomprehension. Ironic detachment makes things worse, not better; the reader-protagonist still wants to test the rules, and detachment invites the frigidity and sophomoric contempt that so often mar computer entertainments. Because the Thespian protagonist is patently unimportant, unheroic, and constrained, the reader's attention is not constantly drawn to the limits of the possible (and hence the shortcomings of the system).

Hypertext, Joyce observed, requires rereading [16] [29]. Thespis can be enjoyed on first reading, but its game-like qualities encourage rereading as well. In the *Trojan Kids,* Alice isn't assigned a mission — there are no captive princesses to be saved — but there are lots of things she could do differently, and many consequences can be imagined. Nothing Alice does can operate by brute force; she can't save Polly by fighting off the Invading Army with her super-powers, she has no spell to redeem Emily nor elixir to cheer Cassandra. But if she had taken Polly for a midnight walk on the beach, perhaps the police would have missed them in the dark? It's worth a try. Ineptly done, this is mere puzzle play; done well, it's the author's expression of the tension between tragedy, where fate is inexorable, and comedy, where our effort[16] can perhaps be rewarded with triumph.

Coherence, causality, and closure — those suspect qualities whose (perceived) absence bedevils the reputation of hyperfiction — can be achieved easily in Thespis if we want them. Assertions form a convenient shorthand for episodes, so explicit temporal dependences are less difficult to maintain. (We can also create Thespian dramas that use multiple decks, one after another: a set of characters and actions becomes a scene or an act of a larger drama) Closure, too, can be achieved in all the conventional dramatic ways.

What becomes of the hypertext patterns with which we have become so familiar? Some flourish unchanged; an assertion that opens up a new topic for discussion introduces a Split, and the retraction then becomes the balancing Join. Indeed, because assertions are easy to retract, Split/Joins may be more common, larger, and more elaborate in Thespian hypertexts. Basements or mirrorworlds can be constructed in space; one room contains the theme, another room the counterpoint, with access between the two restricted by a bottleneck. Feints are at once more problematic (we have no maps) and less (characters lie). And tangles, obviously, are Thespis' natural construct.

Other familiar patterns may be less common in Thespian hypertext. For instance, recurrence in Thespis tends to be rare, brief, and deliberate. Writers who like recurrence can easily provide duplicate actions. The more elaborate cycles that are the staple of many Storyspace hyperfictions are harder to reproduce in Thespis. Douglas' cycle, in which repetition signals closure, is difficult to implement, and Joyce's cycle appears infeasible.

OTHER DIRECTIONS

Thespis is a sketch, a prototype. A host of design decisions are arbitrary. Why does the player choose from *seven* alternatives? Should assertions remain on the blackboard forever, or should they fade over time? When the player sees or hears something, the current Thespis prototype adds the description to the end of a long scroll; would it be better to display the text in a large, dynamic collage [7]? Or in lots of separate windows? Thespis is a chunky hypertext system that generates a smooth (and linear) text; perhaps it should be generating a smooth hypertext, or a chunky one?

It is also interesting to observe that we can add new actions — indeed, entire new characters — to an existing scene. This could give rise to several intriguing possibilities. Not only might we reread a familiar Thespian play, but we might attempt a reading in the presence of a new, supplemental character. We could envision extensible, recombinant fictions, dramas to which readers could add or remove some of the characters. Or we could let several different people control several characters within the same scene, perhaps through the network; notably, Thespis's constraints on actions averts the worst faults of the graffiti problem.

MY FRIEND HAMLET: IMMERSION, GAMES, INTERACTION

Thespis shares some characteristics with Interactive Fiction (IF) — adventure games, MUDs, and MOOs. The resemblance, however, is superficial.

In IF, the reader is the player, the protagonist, the central character. Actions shape the course of events, determining whether how things turn out. In Thespis, the reader is a minor character inhabiting the periphery of the action, a witness to events that unfold. The reader's choices may indeed alter what happens, but the Reader is not the most interesting nor the most active character on stage.

Illusions that place the reader on stage necessarily founder when promised freedom of action is contradicted by the limitations of the simulated environment. IF asks us to find a creative, imaginative, and successful resolution to the dramatic problem. The imaginative reader is bound to think of things the creator never envisioned, and the reader's best thinking inevitably generates the dullest response: "I don't understand." The computational environment can never match our aspirations, and allusions to unlimited computing power of the future (the starship holodeck) can't rectify the fundamental problem: readers will always want to do things nobody (and no computer) could anticipate. That, after all, is why people are interesting, and why we enjoy fiction.

Even if we could experience *Hamlet* on the holodeck, it wouldn't work. Tragedy requires that the characters be blind (as we ourselves, at times, are blind); if you let a sane and sensible reader into the room, everything is bound to collapse. Take Hamlet: it's absolutely obvious that he should go back to school, get roaring drunk, get laid, and await his opportunity. He knows this. Horatio knows this, Ophelia knows this. Even Claudius and Gertrude know — why else send for his college pals? Nobody can bring themselves to say the words — that's the tragedy. But what's to stop the reader? Only brute force and error messages ("You can't do that") that call attention to the arbitrary boundaries of the world. If you make *Hamlet* a game, it has to be rigged.

It's not just Hamlet. Oedipus, for example, needs to get out of town and change his name, to enter the Foreign Legion or the Witness Protection Program. Antigone needs a long talk

[16] Or, in romance, our inherent virtue, our intrinsic wonderfulness. [22]

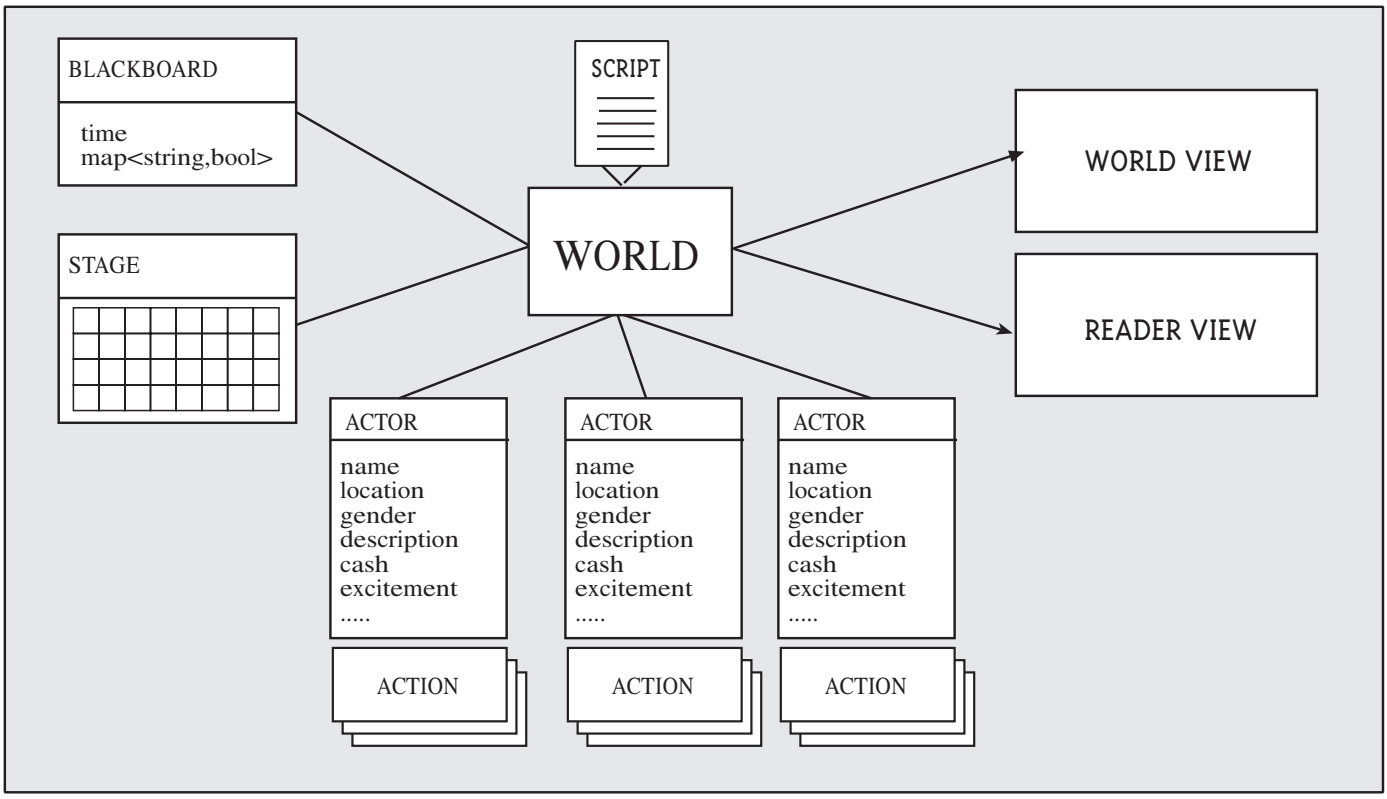

Figure 3. Architectural overview of Thespis.

with her rabbi. Juliet needs to tell her parents exactly what she did last night. She can't, of course, but what's to stop you?

The game is rigged, and constantly calls our attention to deception. Whenever we struggle against the bonds of fate (and the boundaries of the system), we're told, "I don't understand." The more we struggle — the more conviction and intelligence be bring to the action — the greater the likelihood that the system will find no appropriate response. Card Shark avoids this contradiction by foregrounding the familiar convention of reading and drama: we may *want* our favorite characters to prosper, but as readers, spectators, we cannot choose the outcome. Thespis gives us more range of action and might offer us a chance to take a role, but that role is not central and our limitations are evident.

In contrast to the more ambitious interactive fiction projects [23], Thespis makes no real attempt to model character emotions or cognitive state. A trivial mechanism lets the actors choose among possible actions. What really matters is what is said, and everything that can be said in Thespis is written in advance. Authorial control retains its customary (if ambiguous) place[17].

Interactive Fictions tend to be spatial fictions; the implicit narrative of Adventure and Myst is one of travel and discovery[14]. Thespis is performed in imaginative space, too, but Thespian spaces tend to differ in scale and design from Adventurous spaces. IF spaces tend to be numerous, small varied, and thinly populated. Thespis, on the other hand, uses space chiefly as a place in which actors move; Thespian spaces tend to be large, bland, and crowded. There may be interesting settings and props in a Thespian world, but these are static and durable. The actors, on the other hand, are moving and speaking; if we don't listen to them now, we may never hear what they say.

IMPLEMENTATION

Thespis was easy to build. It took less than a person-month to implement. The tools used were not especially suited for fast prototyping: C++ and its standard libraries. The development environment was not especially powerful. Code reuse was limited to a handful of utility classes (smart rectangles and the like) the author habitually carries from project to project. Roughly five work days elapsed from New Project to a build sufficient to enact the opening of *The Trojan Kids.*

Hypertext systems were once major investments, but this is no longer necessarily the case. Documentation, support, and

[17] The reader chooses where to be and what to do, and so may miss interesting things. A special predicate, `MustBeHeard`, identifies actions that may only be performed if the reader can witness them, but its ubiquitous use defeats the purpose of Thespis and it is probably best reserved for crucial revelations. The writer may be discouraged at the thought that the best lines might go unheard; of course, in conventional theater they may be left behind in New Haven or lost in the cutting room.

polish are expensive, of course, but for an experiment they are of little moment.

The script for a Thespian hypertext is composed in a simple markup language[18]. An excerpt should suffice to give the flavor of the current implementation.

```
//introduce a cast member
Actor: {  Name: Emily  X: 5 Y: 3 Gender: f
   Status: 30
   Text: "EMILY is a nice girl, a good
   girl. She is glad the war is over so
   everyone can be friends. She is
   hardworking, plain, and not very
   popular; she was responsible for
   getting the permits and making the
   arrangements for tonight's party."}

//give Emily something to say
Say: { For: Emily Assert: Dreams After: 25
   Text: "I've been dreaming of peace for
   months. For a time when we could just
   have fun again, when the clouds would
   lift. Life goes on, right!

   These shrimp are really good." }

Monologue: { For: Ilana Requires: Dreams
   Text: "I never dream. Or I don't
   remember."}
```

PERFORMANCE

An action's consequences may be both invisible and visible, and even a conspicuous consequence may be invisible if the reader is not able to observe it. Actions do not generate visible consequences directly; instead, they create an object that describes the consequence to one or more *Views*. The view, in turn, determines how to display the consequence. Conversation is audible to the reader if she is within "hearing" range, while other actions may be visible to the reader if the are performed within a larger "vision" range and if the reader's line of sight is not obstructed by walls or obstacles.

Actions are implemented as a family of action objects that encapsulate:

- methods for selecting partners with which to perform the action
- preconditions for starting the action
- preconditions for ending the action
- a description of the performance of the action
- state changes the action imposes on the actor, and on participants.

Actions do not enact their consequences directly; instead, an object describing the action is passed to the World mediator which passes it, in turn, to various *Views* that render the action for the reader. The current prototype's *Views* follow typographic conventions borrowed from the theater; movement, internal monologue, and description are set as stage directions while conversation is typeset as actor speech. The Reader view omits actions the reader cannot see or hear (and abbreviates or omits those that seem unimportant); writers may find a World view useful as well to inspect all actions that take place without regard to the reader's location.

AN OPEN QUESTION

Should Shark and Thespis have been implemented in a modern hypermedia framework such as Construct [27] or FOHM [24]? To do so would almost certainly be feasible, at least in a trivial sense where a custom client (such as the current monolithic prototypes) uses the framework services a repository for opaque data: the client might retrieve an entire context from the middleware . Of greater interest would be dividing functionality between custom middleware services and custom clients. Collaborative functionalities such as Social Shark would be nearly free, and the Thespis client might be simplified be crafting actors as distinct, faceless clients. Whether it is simpler to experiment with a monolithic client or a collection of smaller, intercommunicating objects is, of course, an unsettled question.

Open, collaborative Shark and Thespis would need to retain rather more state information in the session record than seems to be generally envisioned in the OHS literature [31]. The Social Shark Session (or an associated Context) would need to own the deck, dealing cards to players as needed. An interesting synchronization problem appears in Thespis; if actors perform asynchronously, a race condition may arise between two actions, each of which negates a precondition of the other, We could rely on the middleware layer to determine the feasibility of each action, of course, but then network latency delays intrude upon the user interface. For example, if the client represents Card Shark actions as literal cards which the reader moves on the screen, feasibility testing needs to be updated at drag-highlight time.

Local state caching would let the client resolve locally which actions can currently be taken, but introduces opportunities for synchronization faults and complicates the middleware's task of mediation and transaction processing. Maintaining state in the middleware layer could require continuous and fairly costly exchange of complex state objects between middleware and client.

TOWARDS EVALUATION

Card Shark and Thespis present alternative approaches to hypertext. They stand far afield from Storyspace and the

[18] It ought to be XML, but it seemed faster to hand-build a jury-rig parser for these initial experiments. The reader with even a passing familiarity with language design and parser construction may by now be saying, "I could do better than that!" It's true; you could I miss the days when the Hypertext Conference had too many "here is my system" papers. You probably miss them too.

Web, but share the core values of literary hypertext. The first evaluation question is, "can it be done?" Hard upon its heels, comes, "can it be done for a price?" The answer is unambiguously affirmative[19].

A more difficult question, of course, concerns utility: are Shark and Thespis good for anything? Within their intended domain, the question of evaluation is problematic, for it ultimately rests on our judgement of imaginative work created for these exotic systems. Such judgement must await the creation of a body of work. Even then, we might justly question whether success or failure lies within the system or within the work itself.

Thespis was designed for works of the imagination. Can it be used for argumentation, pedagogy, or technical documentation? Clearly sequential presentation is invaluable for mathematical proofs and for some kinds of schoolwork. Information retrieval is clearly ideal for answering specific, well-posed questions. If Thespian hypertext has a place outside the world of imagination, that place most probably lies in exploring multifaceted topics for an expert, engaged audience.

One can readily envision, for example, a lively Thespian discussion about areas of professional and scholarly controversy. Was Captain Cook considered a deity, or merely an unwelcome dinner guest? Is manned space flight cost-effective? Should Web site design emphasize familiarity and ease of use, or strive for unique identity and value? Is there a navigation problem? We might also envision Thespian explorations of naturally discursive subjects — the aesthetics of algorithms, the beauty of chemical synthetic pathways, or the experience of life in London in 1680. Here, the weight of the argument lies in the accumulation of detail and in allowing each reader to find the specific details that speak most powerfully to them. If you want to argue that algorithms or sculpture are beautiful, you'd best be prepared with a variety of examples and let the audience tell you what they like.

Wandering through Thespian spaces lets the reader see what she wants, yet also indicates unobtrusively that there is more to see, and sketches where she might go next.

ACKNOWLEDGMENTS

Diane Greco, Eric A. Cohen, and Charles S. Bennett read early drafts of this paper and made many valuable suggestions. Portions of "My Friend Hamlet" appeared, in different form, in *HypertextNow,* and are used by permission of Eastgate Systems, Inc.

REFERENCES

1. Espen J. Aarseth, *Cybertext: Perspectives on Ergodic Literature*. 1997, Baltimore: Johns Hopkins University Press.

2. Mary-Kim Arnold, *Lust.* Eastgate Quarterly Review of Hypertext, 1993. **1**(2).

3. Mark Bernstein, *Deeply Intertwingled Hypertext: The Navigation Problem Reconsidered.* Technical Communication, 1991. **1991**: p. 41-47.

4. Mark Bernstein, *The Navigation Problem Reconsidered*, in *Hypertext/Hypermedia Handbook*, E. Berk and J. Devlin, Editors. 1991, McGraw-Hill: New York. p. 285-297.

5. Mark Bernstein, *Conversations With Friends: Hypertexts With Characters*, in *Hypermedia Design*, S. Fraïse, *et al.*, Editors. 1995, Springer: London. p. 207-215.

6. Mark Bernstein. *Patterns of Hypertext*. in *Hypertext '98*. 1998. Pittsburgh, PA: ACM.21-29

7. Mark Bernstein. *More Than Legible: On Links that Readers Don't Want to Follow*. in *Hypertext 2000*. 2000. San Antonio, Texas: ACM.216-217

8. Sven Birkerts, *The Gutenberg Elegies.* 1994, Boston: Faber and Faber.

9. Jay David Bolter and Michael Joyce. *Hypertext and Creative Writing*. in *Hypertext '87*. 1987. Chapel Hill: ACM.41-50

10. M. D. Coverly, *Califia*. 2000, Watertown, MA: Eastgate Systems, Inc.

11. Jane Yellowlees Douglas, *The End of Books -- or Books without End?* 2000, Ann Arbor: University of Michigan Press.

12. John Gardner, *The Art of Fiction: Notes On Craft for Young Writers.* 1983, New York: Vintage Books.

13. Gene Golovchinsky and Catherine C. Marshal. *Hypertext Interaction Revisited*. in *Hypertext 2000*. 2000. San Antonio, Texas: ACM.171-180

14. Henry Jenkins and Mary Fully, *Nintendo and New World Narrative*, in *Communications in Cyberspace*, S. Jones, Editor. 1994, Sage: Los Angeles.

15. Michael Joyce, *afternoon, a story*, . 1990, Eastgate Systems, Inc.: Watertown, MA.

16. Michael Joyce, *Of Two Minds:Hypertext Pedagogy and Poetics*. 1994, Ann Arbor: University of Michigan Press.

17. Michael Joyce, *Othermindedness: the emergence of network culture*. 1999, Ann Arbor: University of Michigan Press.

18. David Kolb. *Scholarly Hypertext: Self-Represented Complexity*. in *Hypertext 97*. 1997. Southampton, U.K.: ACM.29-37

[19] I did not know, a month ago, what the answer would be. I delayed starting for about ten months, suspecting the cost would exceed my meager budget.

19. George P. Landow, *Hypertext 2.0: The Convergence of Contemporary Critical Theory and Technology, 2nd edn.* 1997, Baltimore: Johns Hopkins Press.

20. George P. Landow and Jon Lanestedt, *The In Memoriam Web*, . 1993, Eastgate Systems, Inc.: Watertown MA.

21. B. Laurel, *Computers as Theatre*. 1991: Addison-Wesley.

22. David Mamet, *Three Uses Of The Knife: on the nature and purpose of drama*. 1998, New York: Columbia University Press.

23. Michael Mateas and Andrew Stern. *Towards Building a Fully-Realized Interactive Drama.* in *Digital Arts and Culture*. 2001. Providence, RI

24. Dave M. Millard, *et al. FOHM: A Fundamental Open Hypertext Model for Investigating Interopability Between Hypertext Domains.* in *Hypertext 2000*. 2000. San Antonio, Texas: ACM.93-102

25. Laura Miller, *www.claptrap.com*, in *New York Times Book Review*. 1998: New York. p. 43.

26. Janet Murray, *Hamlet On The Holodeck: The Future of Narrative in Cyberspace*. 1997, New York: The Free Press.

27. Peter J. Nürnberg, *et al.*, *A component-based open hypermedia approach to integrating structure services.* New Review of Hypermedia and Multimedia, 1999. **5**: p. 179-206.

28. H. VanDyke Parunak, *Don't Link Me In: Set Based Hypermedia for Taxonomic Reasoning*, in *Hypertext'91*. 1991, ACM Press: San Antonio. p. 233-242.

29. Anja Rau, *Wreader's Digest: How To Appreciate Hyperfiction.* Journal of Digital Information, 2000. **1**(7).

30. B. Reeves and C. Nass, *The Media Equation - How People Treat Computers, Television , and New Media Like Real People and Places.* 1996, Cambridge, MA: Cambridge University Press.

31. Siegried Reich, *et al.*, *Addessing interoperability in open hypermedia: the design of the open hypermedia protocol.* New Review of Hypermedia and Multimedia, 1999. **5**: p. 207-246.

32. Mitchell Resnick, *Turtles, Termites, and Traffic Jams: Explorations in Massively Parallel Microworlds*. 1997, Cambridge: MIT Press.

33. P. David Stotts and Richard Furuta, *Petri-net based hypertext: Document structure with browsing semantics.* ACM Transactions on Office Information Systems, 1989. **7**(1): p. 3-29.

34. Stephanie Strickland, *True North*, . 1998, Eastgate Systems, Inc.: Watertown, MA.

35. Susana Pajares Tosca. *A Pragmatics of Links*. in *Hypertext 2000 Proceedings*. 2000. San Antonio, Texas: ACM.77-84

36. Jill Walker. *Piecing together and tearing apart: finding the story in afternoon.* in *Hypertext '99*. 1999. Darmstadt, Germany: ACM.111-118