

Aaron Reed

Knock's Memories: Creating an Emotional Story Space With Inform 7

“He’d caught some disease in the womb that forced him to tell stories. The weight of those stories bowed his legs and bent his spine a bit.”

— *Sherman Alexie, Reservation Blues*

1 Introduction

Good storytellers hold universes in their heads. From erudite novelists to grizzled old men around campfires, every person who tells a story creates a mental image of characters, props for them to use, places for them to inhabit and events for them to experience. Human storytellers carefully select the words that best reveal to their audience the state of the world in their imaginations. To teach a computer with no imagination how to tell a story, we must first explain to it our story world—the places, things, people, and plans that make up a narrative. Only then can the computer select appropriate words to explain that story to an audience sitting at a keyboard, and only then can it understand what the audience types back.

Inform 7¹, first publicly released in 2005, is today the preeminent design system for interactive fiction (IF). Its lineage can be traced in a fairly straight line all the way back to Zork, the first interactive fiction released by Infocom. While prior versions of Inform were “a computer programmer’s tool which aimed to be welcoming to creative writers,” Inform 7 “aspired to be the other way around”²: a tool for making interactive stories designed first and foremost for writers,

¹<http://www.inform7.com/>

²Graham Nelson, 2005. “Natural Language, Semantic Analysis and Interactive Fiction.” <http://www.inform->

not coders.

Inform 7 uses a natural language syntax that lets authors create story worlds with straightforward English sentences, which its creator calls “a radically humanising interface for the writing of interactive fiction.” Inform lets us use declarative sentences in natural-sounding English to define a story world. These sentences can be as simple as:

Tom is a person.

...but can also be much more complicated:

Instead of attacking Tom when something lethal is held, now every nearby watchdog owned by Tom hates the player.

The language also allows for inline logic, which makes complex conditional text generation easy to produce:

Every turn when a container (called the sack) held by someone visible (called the unlucky holder) is bursting, say “[The sack] splits and breaks under the weight! [if the player is the unlucky holder]You discard[otherwise][The unlucky holder] discards[end if] its ruined remains, looking miserably down at [the list of things in the sack] on the floor.”

By comparison, the equivalent code in Inform 6, the language’s immediate predecessor, reads as follows:

Initialise [;

fiction.org/I7Downloads/Documents/WhitePaper.pdf

```

    ! start a daemon for every sack object in game

    objectloop (s ofclass Sack) {
        StartDaemon(s);
    }
];

Class sack
    with daemon [ unlucky_holder;

        ! check to see if sack is bursting and its owner is visible

        unlucky_holder = parent(self);

        if ((self.bursting == 1) && TestScope(unlucky_holder, player)) {
            print (The) self, " splits and breaks under the weight! ";
            if (unlucky_holder == player) {
                print "You discard";
            } else {
                print (The) unlucky_holder, " discards";
            }
            print " its ruined remains, looking miserably down at ";
            WriteListFrom(child(self), DEFART_BIT + ENGLISH_BIT);
            print " on the floor.^";
        }

    ],

    has container;

```

Both versions produce identical games, but the first is easier to understand; the parity

between the language of the source code and the language of the story's output makes creating story structures in Inform 7 incredibly naturalistic.

To further demonstrate this delightful symmetry, in this paper we'll build part of the workings of an interactive story called *Sand-dancer*, the example game for a forthcoming book on Inform 7, which encodes the main character's emotional journey. In the process we will gloss over detailed explanation of Inform 7's syntax in order to provide a broad overview of the capability and feel of working in the language.³

2 A quick overview of *Sand-dancer*

Sand-dancer is a short interactive story that tells the story of a surreal night in the life Nakaibito "Knock" Galvano, an eighteen-year-old with a pregnant girlfriend and a dead-end job. As the narrative begins, Knock has just driven off the road and crashed his pickup truck, somewhere on a lonely New Mexico highway between work and home. The night is supernaturally dark, and Knock can't seem to find the highway again, so he explores the abandoned outbuildings of an old electrical substation looking for a way to either fix his truck or find shelter to spend the night in the desert. As Knock explores, memories of his troubles keep bubbling to the surface of his mind, and he can't help but brood about the direction his life is going.

But as the evening progresses, stranger and stranger things begin to happen. An old radio that at first seems to connect him to a helpful BLM dispatcher starts to spout strange and surreal invective. Objects seem to sometimes disappear and rooms are not always where Knock thought

³This paper is based in part upon material from the author's book *Creating Interactive Fiction with Inform 7*, Cengage, New York, 2010.

he left them. Soon he encounters a series of visions, spirit animals who seem to combine elements of all the conflicting mythologies he's been brought up with: the New Age mysticism of his white grandmother, second-hand stories of his half-remembered Navajo father, and the cultural impositions of his Mexican stepfather. As Knock's reality continues to soften, the night culminates in a visit with the most powerful and sinister spirit animal of all: the giant lizard Sand-dancer, who gives him a disturbing choice between returning to his normal life and an escape that seems too good to be true.

Sand-dancer is strongly plotted, without many forking branches: it is certainly closer to pure story than pure game. But it still touches on several strengths of IF: exploration, a mature story world, and meaningful choice. Throughout the game the player makes choices (which memories to retrieve, which goal to work for, what attitude to take with the spirit animals) and these mirror the game's ultimate resolution.

3 Defining Memories

The central component of our system for revealing Knock's backstory is the memory. Certain objects in the story world are linked to memories; as the player finds these objects, they trigger specific memories from Knock's past. These memories can then be later selected by the player to be brooded about, reliving their contents.

Inform calls classes *kinds*. The base kind which represents all objects present in the story world is the *thing*. Therefore we define memories with:

A memory is a kind of thing.

Inform includes a set of instructions called the Standard Rules that encode many assumptions about the nature of IF stories. The Standard Rules divide space into discrete *rooms* and define moving between them via cardinal directions; they define notions of enclosure such that each character or prop can only be in one room at a time, and that certain props called *containers* or *supporters* can enclose other props; rules such as containers being unable to contain themselves; and many other features commonly included in a reality-based story world.

But real stories inevitably define more than just spatial relationships between their components: plans and the steps needed to complete them; hierarchies of material value or aesthetic judgment; complex relationships between members of a family or friendship group. To model these more abstract but richly detailed connections, Inform gives us an extremely powerful tool called *relations*.

A relation defines one way in which the rooms, things, or properties in a story world connect to each other. To connect our memories to certain objects in our game world, we use relations. Here's how we introduce this concept to Inform.

Suggestion relates various things to one memory (called the vision).

“Suggestion” is the arbitrarily-chosen name of this relation, and the second half of the sentence explains the nature of the connection. In this case, we’re saying that any number of things can each suggest exactly one other thing, which must be of the kind memory. We had to define our initial concept a little more precisely in order to create this line: if we wanted things to be able to suggest more than one memory each, we might have said *various things to various memories*, and we’ve also implied that it’s okay for multiple things to suggest the same memory (otherwise we’d have said *one thing to one memory*). Also implicit in the *various to one* relation

is that the connection only works in one direction: this distinction prevents us from mistakenly declaring that a memory can suggest a thing.

The parenthetical clause (*called the vision*) allows us to refer to the relation from the other direction. If we've defined that the rusty tin can suggests the fights on the playground, Inform knows that the fights are the vision of the rusty tin can.

We've defined the nature of our suggestion relation, but we don't have a way yet to connect objects with it. In the same way Inform code uses built-in verbs like "is," "holds," "contains," and even "relates," we need to teach Inform a verb that will let us state that things are related by suggestion. Here's how we do it:

The verb to suggest (he suggests, they suggest, he suggested, it is suggested, he is suggesting) implies the suggestion relation.

Due to the vagaries of natural language (and English in particular), Inform doesn't try to guess the different forms of speech a newly defined verb might take. It doesn't look so difficult in this case, but imagine even a fairly innocuous verb like "to become": Inform would be hard-pressed to know that "became" or even "becoming" (with the dropped "e") were different conjugations of the same word.

Once we go through this exercise, Inform can understand our use of the relation no matter what context we're talking in:

The last day of high school is a memory. It is suggested by the guidebook.

Which is exactly equivalent to:

The guidebook suggests a memory called the last day of high school.

It's important to note that relations are purely abstract. The above sentence alone does not imply anything about what happens when things suggesting memories are encountered. Relations simply define how the important things in a story world are conceptually connected.

4 Emotional Baggage

Memories will be the primary way we reveal the story of Knock's past, but they'll also serve a critical role in interactions with the spirit animals as the player barter with them for talents, especially in the pivotal final scene with Sand-dancer.

A memory is an abstract concept, but in order for players to perform actions with one, they normally need to be able to see it. While we could use advanced trickery to make the relevant memories accessible at the appropriate time (using Inform's concept of *scope*), a more direct approach, and one that fits *Sand-dancer's* somewhat whimsical style, is for Knock to simply carry any memories he's uncovered in his inventory. Let's create an appropriate container for these carried memories:

The player carries an open transparent unopenable container called emotional baggage. The description is "Your guidance counselor used to say you're always carrying it with you. You imagine it's kind of ugly, lumpy, and green, and definitely has a stuck zipper.[if at least one thing is in baggage][paragraph break]Plenty to brood about in there, like [the list of things in baggage].[end if]".

With this, we might see transcripts like the following:

>INVENTORY

You are carrying:

a pack of cigarettes

a piece of jade

some emotional baggage

the last day of high school

grandmother's stories

your shit job

your keys

>EXAMINE BAGGAGE

Your guidance counselor used to say you're always carrying it with you. You imagine it's kind of ugly, lumpy, and green, and definitely has a stuck zipper.

Plenty to brood about in there, like the last day of high school, grandmother's stories, and the day you met Ocean.

Of course, normal actions like taking and dropping shouldn't work on this purely metaphorical item:

Instead of doing anything other than examining to emotional baggage: say "It's not real, bro."

And we should also account for things being indirectly done to the baggage:

Instead of inserting anything into emotional baggage: say "It's not real, bro."

Whenever the player encounters an item that triggers a memory, we move that memory into the emotional baggage. Since Knock is always carrying it and it's always open, the memory will then be visible and thus available to relevant commands at any point afterwards in the story.

Definition: a thing is charged if it suggests a memory which is not in emotional baggage.

Every turn when a charged thing (called the item) is visible: move the vision of item to emotional baggage; say "Something about [the item] reminds you of [the vision of item]."

5 Brooding

Once a memory has been uncovered by finding a thing that suggests it, how does the player retrieve the memory? There's nothing wrong with a simple solution here, either: we can just put the memory's description into its description property (used by Inform to respond to the command EXAMINE).

But EXAMINE seems like an odd command to type to relive memories. Let's give players a custom verb that acts as a targeted synonym.

Brooding is an action applying to one thing. Understand "brood about [something]"

as brooding.

Check brooding when noun is not a memory: say "Eh. You couldn't really get into a good brood about that when it's this damn cold." instead.

Carry out brooding: say "[description of noun][line break]".

Instead of doing anything other than brooding to a memory: say "As if. All you can really do is BROOD ABOUT it."

This hints and allows for the player to type commands like:

```
>BROOD ABOUT MY SHIT JOB
```

... which nicely encapsulates the direction the genre has moved in since the days of Zork.

Now that we've got our infrastructure in place, let's create some memories. While most players will probably brood about a good chunk of these, we have no way of guaranteeing that a given player will see any given memory, so none of them should contain any vital piece of information necessary for advancing the plot: instead, each memory should add to the picture of Knock's character, like pieces of a puzzle.

The memories are our way to reveal the player-character's backstory, and to build up to the revelation near the end of the game that Knock's biggest problem is his pregnant girlfriend. We try to capture something of his personality and motivations in the few paragraphs set aside for each one.

The last day of high school is a memory. It is suggested by the guidebook. The description is "You didn't know it was going to be your last day. But that morning you got called in to the principal's office and fat bald Mr. Cox and pissy old Ms. Burke were there, and they looked kind of like strong animals stalking weak animals, and you knew something bad was up.

I have here Mr. Morales (and god damn do you hate being called that) a test you took last week in Ms. Burke's sophomore English class. Questions have been raised (and he looked up at Ms. Burke like he was trying to pass the buck) questions have been raised about the quality of your essay, and whether a student with your academic and behavioral record (he scratched his bloated nose meaningfully) could have plausibly produced such an essay, and you get the idea. They thought you cheated.

No. They knew you cheated, deep in their smug empty hearts. They wanted you to admit it, say you were a cheater and a liar. But you weren't. You wrote that essay, every god damn word, because you really really liked the book for once and wanted to show Mrs. Burke that maybe if they gave people better books to read kids would actually learn something. But they wanted a confession. They wanted a thieving example they could parade in front of the school. Someone of your academic and behavioral and economic and racial background and yeah, screw this shit. So you got up and left and never came back. Drop out, hell. You walked out and you'd do it again."

Sand-dancer has twelve memories that can be discovered in all, but let's just add one more representative example.

A piece of jade is in your pickup truck. A memory called grandma's stories is suggested by the piece of jade.

The description of grandma's stories is "There are dark spirits who roam the earth, little Knock. Grandma used to say that, holding you tight and stroking your hair. There are dark spirits who roam the earth, but you're not alone. Oh, no. I'm here. (She'd kiss your head and you'd squeeze her back.) But others are watching out for you too. You have three animal guardians, hmm? Spirits who are always watching over you. Oh, you can't always trust them to know what's best. Remember that, Knock. But when you need help, they'll come, and protect you from the worser things in the world.

Mom would yell at Grandma a lot for filling your head with that new-age bullshit.

Grandma grew up white and midwestern and Baptist, but had started wearing things with feathers and playing the pan flute by the time you were born. She seemed to really like having a son-in-law who was Native American or American Indian or Indigenous Peoples or whatever she'd decided the right term was that week, and she was pretty pissed when mom left him. Anyway. Her stories were mostly BS, you guess, but some of them stuck with you. When it's dark you still wonder if your spirit animals are out there somewhere, and what the hell is taking them so long to find you."

6 Talents

In fact, Knock's spirit animals will visit him sooner than he thinks. During three pivotal scenes in the story, the player encounters a sometimes mystical, sometimes irreverent entity who offers him a talent he can use to advance further towards his goal (either fixing his truck or surviving the night).

A spirit animal is a kind of person. Hare, Coyote, and Sand-dancer are male spirit animals.

Each spirit animal offers two talents, but the player can only choose one for Knock to acquire. Certain talents help advance the player towards one of the two goals, emphasizing the theme that Knock's future options are growing limited, and going through one door will irrevocably close another.

A talent is a kind of thing. strength, courage, luck, scent, honor, spirit, and freedom are talents. *[Note the uncapitalized "strength": we don't want the talents to have articles, but we also don't want them to be capitalized like proper nouns.]*

Hare carries strength and courage. Coyote carries luck and scent. Sand-dancer carries honor, spirit, and freedom.

We keep track of which talents the player has chosen by simply moving them to Knock's inventory: again, this fits the style of the surreal story we're telling. Acquired talents can't otherwise be interacted with by the player:

Instead of doing anything to a held talent: say "It's part of you, now."

Checking whether various things are possible for Knock to do is then simply a matter of seeing whether he holds the required talent. Here's an example, from one of the puzzles related to fixing the pickup truck:

Instead of taking something enclosed by the hole when player does not hold courage:

say "You reach your hand towards the hole, then pull back with a start as you brush cobwebs. Nightmare visions of fat black spiders shudder through your brain. No way are you reaching in there."

After taking duct tape: say "You reach down into the hole, brushing the cobwebs away impatiently, and pull out the duct tape."; remove cobwebs from play. *[Note that this rule only runs if the instead rule above does not.]*

7 Trading

One additional wrinkle: the spirit animals do not just give away the talents they offer. They demand payment in return, and the only thing Knock has that is of value to them are his memories. Hare requires just one memory in trade; Coyote needs two; and Sand-dancer's price for the most coveted item is very high, indeed.

A spirit animal has a number called price. The price of Hare is 1. The price of Coyote is 2. The price of Sand-dancer is 3.

To barter with memory, we need another new verb:

Trading is an action applying to two things. Understand "trade [something held] for [something]" as trading.

Since the person to be traded with is not directly invoked in the command, we need an action variable to keep track of who this is.

The trading action has a person called the person traded with. Rule for setting action variables for trading: now the person traded with is a random visible spirit animal.

We need a series of check rules to dictate when this action is allowed to happen: only when a spirit animal is present, only with the proper components, and not for anything that's already been acquired.

Check trading when the person traded with is not a spirit animal: instead say "[The noun] can't be traded with."

Check trading when the noun is not a memory: instead say "'I'm only interested in trading memories,' says [the person traded with]."

Check trading when the second noun is not a talent: instead say "'I only have talents to offer you,' says [the person traded with], '[list of visible unheld talents].'"

Check trading when the second noun is held: instead say "You already have the talent of [second noun]."

Inform's scope rules ensure that the player can't trade for talents that aren't available (i.e., aren't held by a visible spirit animal.)

If the action is successful, we check to see if the person traded with has received a fair price. If so, we complete the transaction and move the items involved to their proper homes.

Carry out trading:

move the noun to the person traded with;

let cost be the price of the person traded with;

if the person traded with holds at least cost memories:

move the second noun to the player;

say "[The person traded with] nods. 'A fair trade,' he says, and something happens inside you. [The noun] is gone— no, that's not it. It's just changed. Not as heavy. Shifted, transmuted into something else ... yeah, into [the second noun]. Cool.'";

otherwise:

say "[The person traded with] nods. 'Yes,' he says, 'that is an acceptable trade; but it is not yet enough.' You feel [the noun] slip from the forefront of your mind into strange, unplumbed depths."

Knock's experience (which, perhaps, is all just a crash-induced hallucination) is helping him to process his unresolved memories. He's turning them into useful attributes that will make

him better able to solve his problems.

8 Sand-dancer's Price

Sand-dancer, unlike the weaker spirit animals, offers Knock a choice of three talents, not just two.

Sand-dancer reveals that the rarest of all talents, freedom, will solve all of Knock's

problems—even the biggest—but the price is every one of his memories.

Carry out trading when the second noun is freedom:

move the noun to the person traded with;

if the player holds at least 1 memory:

say "Sand-dancer swallows the memory whole. 'More,' he says, tongue

licking the air greedily.";

otherwise:

perform the unmaking ending.

In the unmaking ending, the story concludes with a disturbing revelation: Sand-dancer has solved Knock's problems too well. His girlfriend's baby has been unmade by unmaking the father. As Knock watches his life unspool in reverse at an ever-increasing velocity, he realizes the lizard has pushed him off the cliff of now:

"You're unmeeting Ocean and unditching school and unsmoking for the first time behind the dumpsters, faster and faster, stomach in your throat, memories blowing back your hair as they rocket past. You're a kid again unskateboarding and unlearning video game combos, screaming backwards faster and faster and you get

it now, you get the lizard's joke. He's making you free by unmaking you, fixing the burden by erasing the guy who's bearing it, and you're unlearning to read and unlearning to walk and uncrawling and uncrying and then some brilliant moment of light and noise and chaos comes shrieking towards you hella fast, fast, faster than anything and it's too late, you're crashing, you're crashing, you crash..."

The other two endings hold more hope for Knock. With the more modest talent of honor, the player is able to fix his pickup truck, drive home to his girlfriend, and do what he's decided is right: propose to her and start a family, no matter how much is stacked against them. Choosing spirit, on the other hand, helps Knock hole up through the chilly night till a ranger comes to pick him up in the morning. Asking to be dropped off in the next town over, Knock remakes his life as the rootless wanderer he was always destined to become.

9 Conclusion

Inform 7's natural language syntax allows for stories with mature themes to be told through mechanics that are easily represented and understood through straightforward English sentences. Stories like *Sand-dancer* are simply not told in the mainstream gaming industry, and even the experimental and indie scenes would struggle to find an effective graphical representation of Knock's trading of memories for innate talents.⁴

Interactive fiction is at its best when it embraces its literary nature as a strength, not a weakness. After three decades of maturation, the medium continues to produce fascinating new work. Inform 7's unique structure pushes IF in new directions by shrinking the distance between

⁴The complete *Sand-dancer* source text and playable story is available at <http://sand-dancer.textories.com/>.

designer and storyteller a little more.