Procedural Authorship: A Case-Study Of the Interactive Drama *Façade*

Michael Mateas
School of Literature, Communication & Culture and
College of Computing, Georgia Inst. of Technology
michaelm -at- cc.gatech.edu

Andrew Stern
Procedural Arts
andrew -at- proceduralarts.com
www.interactivestory.net

(co-authors listed alphabetically)

ABSTRACT

The essence of the computer as a representational medium is procedurality – the ability of the computer to engage in arbitrary mechanical processes to which observers can ascribe meaning. Taking full representational advantage of the computer thus requires procedurally literate authorship, that is, artists and writers who are able to think about and work within computational frameworks; in the extreme case of developing new modes of computational expression, authors must be highly proficient in the use of general purpose programming languages. We examine issues of procedural authorship using the interactive drama *Façade* as a case study. *Façade's* explicit design goal is to provide the player with local and global agency over the evolution of the dramatic experience; this requires a level of procedurality previously not implemented in interactive narrative.

1. INTRODUCTION

The essence of the computer as a representational medium is procedurality – the ability of the computer to engage in arbitrary mechanical processes to which observers can ascribe meaning. Computers do, of course, participate in the production of imagery, support communication between people via the mediation of long-distance signals, control electro-mechanical devices, and support the storage and interlinking of large quantities of human-readable data. Many tools are available that allow users to engage these various capacities of the computer, such as image manipulation or webpage authoring, without requiring users to think procedurally. But it is precisely the computer's ability to morph into these special purpose machines that highlights the computer's procedural nature. These special purpose machines (e.g., tools) are made out of computational processes; the computer's ability to engage in arbitrary processes allows it to morph into arbitrary machines.

Taking full representational advantage of the computer thus requires procedurally literate authorship, that is, artists and writers who are able to think about and work within computational frameworks. By procedural literacy, we mean the ability to read and write processes, to engage procedural representation and aesthetics, to understand the interplay between the culturally-embedded practices of human meaning-making and technically-mediated processes. Even for new media practitioners who don't themselves write much code, procedural literacy is necessary for successfully participating in interdisciplinary collaborative teams, and for understanding the space of possibility for digital works. Many authors find themselves engaged in some level of programming, especially for interactive work which, of necessity, requires conditional response to input, and thus the specification of a process. In the extreme case of developing new modes of computational expression, authors must be highly proficient in the use of general purpose programming languages, used to construct new languages and tools specialized for the new representational mode.

In this paper we provide a case-study, using the interactive drama *Façade*, of this last case of procedural authorship. *Façade* represents a new mode of computational representation, *interactive drama*, combining the game-like pleasure of moment-by-moment interaction with believable characters, with the story-like pleasure of participating in and influencing a long-term dramatic progression. As procedural authors, we undertook several design-plus-programming tasks: deconstructing a dramatic narrative into a hierarchy of story and behavior pieces; designing an AI (artificial intelligence) architecture, and collection of special purposes languages within the architecture, that respond to and integrate the player's moment-by-moment interactions to reconstruct a real-time dramatic performance from those pieces; and writing an engaging, compelling story within this new framework.

This paper makes a case for the importance of procedural authorship, describes the design goals of *Façade* and how these goals could only be met through a highly procedural approach to interactive narrative, and finally describes *Façade's* architecture, content organization, and the experience of authoring within this framework.

2. PROCEDURALITY

Murray identified four essential properties of the computer as a representational medium: that computers are procedural, participatory, encyclopedic and spatial [17]. The procedural, of course, refers to the machinic nature of computers, that they embody complex causal processes, and in fact can be made to embody *any arbitrary* process. The participatory refers to the interactive nature of computers, that they can dynamically respond to outside signals, and be made to respond to those signals in a way that treats those signals as having the meaning ascribed to them by people (that is, non-arbitrary response). The encyclopedic refers to the vast storage capacity of digital computers, and their ability to organize, retrieve and index stored material. The spatial refers to the ability of digital computers to represent space, whether that is the physical space of virtual reality and games, or the abstract space of networks of information.

Various communities of practice tend to hold different properties as central. Here we provide a few examples of the privileging of various properties. For the Demoscene, a largely competition-

Mateas, M. and Stern, A. Procedural Authorship: A Case-Study of the Interactive Drama Façade. In *Digital Arts and Culture: Digital Experience: Design, Aesthetics, Practice (DAC 2005)*, Copenhagen, Denmark, 2005.

oriented subculture with groups and individual artists competing against each other in technical and artistic excellence [18], procedurality is central; the aim is to procedurally generate as rich an audio-visual experience as possible using the minimum amount of stored content. The participatory is privileged in rhetorics of agency, control, and co-authorship, and has been adopted by communities as diverse as user-interface design, interactive art, and digital marketing. Database art privileges the encyclopedic, sometimes viewing all new media art practice as metaphorically related to the manipulation and resequencing of data stores. Spatiality is privileged by such diverse communities as virtual reality, game design, and hypertext.

While all of these properties play some role in various computational media, procedurality is the essential, defining property of computational media, without which the other properties could not exist.

Any participatory system requires the specification of potential action that is carried out in response to a stimulus. Capturing a space of potential action requires specifying a machine or process that can actualize the potential under different contingencies. In other words, participatory systems require procedurality. The converse is not true; there can be procedural systems that are not participatory, but rather execute a fixed process without accepting input. Many generative art systems, such as Aaron [10], exhibit procedurality without being participatory.

Encyclopedic systems are similarly dependent on procedurality. Without the ability to perform operations on data, to be able to access, re-sequence, search, modify, index and so forth, large data stores are useless. Without the procedural competencies of web search technologies, for instance, the web could literally not exist in its current scale. There would be no reason to create a new web page without the ability to relate the page to other, already published pages, and the ability for others to be able to find and view your page. Again, the converse is not true. Processes can create elaborate experiences from very small kernels; this capability is in fact the inspiration for the Demoscene.

The spatial is clearly a derivative property, a representational illusion actively maintained by a process. Graphical spatial representations make use of procedural models to compute and dynamically update the displayed space. Interactive spaces, which create the sense of space by supporting active navigation through the space and may not make use of 2D or 3D graphical representations of the space at all, depend on the participatory, which in turn is supported by procedurality.

The goal here is not simply to play a dominance game between the various representational properties of computers, but to avoid serious confusions and misunderstandings that can arise in new media theory and practice from misunderstanding the central importance of procedurality. Without a deep understanding of the relationship between what lies on and beneath the screen, scholars are unable to deeply read new media work, while practitioners, living in the prison-house of "art friendly" tools, are unable to tap the true representational power of computation as a medium.

Without an understanding of procedurality, of how code operates as an expressive medium, new media scholars are forced to treat the operation of the media artifacts they study as a black box, losing the crucial relationship between authorship, code, and audience reception. Code is a kind of writing; just as literary scholars wouldn't dream of reading only translated glosses of work, never reading the full work in its original language, so new media scholars must read code, not just at the simple level of primitive operations and control flow, but at the level of the procedural rhetoric, aesthetics and poetics encoded in a work.

New media practitioners without procedural literacy are confined to producing those interactive systems that happen to be possible to produce within existing authoring tools. To date, such tools tend to have an encyclopedic orientation; in the absence of significant support for procedural authorship (i.e. programming), authorship consists of the gathering together of numerous media assets (video, sound, text, image, etc.), and the spatial and temporal composition of those assets within the procedural framework supported by the tool (e.g. linking). This approach fundamentally limits the size and complexity of new media artifacts. For interactive works, this problem is especially severe, as it forces the author to pre-specify and explicitly author responses to all possible interactive situations.

2.1 Procedurality and Content

To describe the relationship between computation and media assets, Chris Crawford introduced the term *process intensity* [4]. Process intensity is the "crunch per bit", the ratio of computation to the size of the media assets being manipulated by the system. If a game (or any interactive software) primarily triggers media playback in response to interaction, it has low process intensity. The code is doing very little work – it's essentially just shoveling bits from the hard drive or CD-ROM to the screen and speakers. As a game (or any interactive software) manipulates and combines media assets, its process intensity increases. Algorithmically generated images and sound that make no use of assets produced offline have maximum process intensity.

Process intensity directly enables richness of interactivity. As process intensity decreases, the author must produce a greater number of offline assets (e.g. pre-rendered chunks of text, animations or video) to respond to the different possible interactions. The number of offline assets required to maintain a given level of interactivity increases exponentially as process intensity decreases; therefore, in general, decreases in process intensity result in decreases in the richness of interactivity.

Though games have a relatively high process intensity within the space of new media artifacts, contemporary games are pushing against authoring limits caused by an over-reliance on non-procedural, static assets. Contemporary games such as Electronic Arts' *The Lord of the Rings* franchise currently contain more media files than lines of code. Even open-world games such as the *Grand Theft Auto* franchise, lauded for their simulated, procedural worlds, still use static assets for every vehicle, every type of person, every building, every weapon, and so forth.

Furthermore, developers at a recent Game Developers Conference voiced concern that next-generation console game hardware will only exacerbate this content crisis. The requirement for ever-more detailed graphics to entice consumers to purchase next-generation consoles means that assets become more expensive to produce, requiring ever larger teams, making games more expensive. Consumers want more gameplay, meaning larger games, thus requiring even more assets to be produced; this all results in a

positive-feedback loop that is considered by many to be unsustainable.

Where insufficient procedurality is creating a crisis in the authoring of traditional games, it has prevented some long sought-after genres of interactive art and entertainment, such as the high-agency interactive story, from even getting off the ground. Bringing process intensive, AI-based techniques to the problem of interactive story was one of the fundamental research goals of our interactive drama, *Façade*.

3. THE INTERACTIVE DRAMA *FAÇADE*

3.1 A Case-Study for Procedural Content

Motivated by our belief that the research into highly procedural authoring methods will enable yet-to-be-realized genres of interactive art and entertainment, we undertook the development of the interactive drama *Façade* [7]. The dream of interactive drama, perhaps best envisioned by the *Star Trek* Holodeck and first presented in an academic context by Brenda Laurel in *Computers as Theatre* [8], has players interacting with compelling, psychologically complex characters, and through these interactions having a real influence on a dynamically evolving storyline. Using a decade of prior research from the Carnegie Mellon Oz Project [2, 9] as a starting point and our belief that a fully-realized interactive drama had not yet been built, we embarked on a five year effort to develop procedural authoring methods for believable characters, natural language conversation, and dynamic storyline, integrated into a small but complete, playable experience. Publicly released in July 2005, *Façade* has been downloaded over 100,000 players worldwide as of this writing, and received widespread critical acclaim [6].

Figure 1. Grace and Trip in *Façade*, viewed from the player's first-person perspective.

Enjoyable video games tend to be highly procedural in implementation, because among implementation methods, procedurality affords the greatest degree of dynamism and reactivity – features very satisfying to players. The best procedural video games excel at giving players high-agency experiences, that is, providing ample opportunities for the player to take action and receive immediate feedback. With *Façade* we wanted to create an interactive drama that provides the level of immediate, moment-by-moment agency, i.e. *local agency*, found in games. But unlike games, we want the player to experience *global agency*, that is, longer-term player influence on the overall story arc, over which topics get brought up, how the characters feel about the player over time, and how the story ends.

Like contemporary games, *Façade* is set in a simulated world with real-time 3D animation and sound, and offers the player a first-person, continuous, direct-interaction interface, with unconstrained navigation and ability to pick up and use objects. But like drama, particularly theatrical drama about personal relationships such as *Who's Afraid of Virginia Woolf?* [1], *Façade* uses unconstrained natural language and emotional gesture as a primary mode of expression for all characters, including the player. Rather than being about saving the world, fighting monsters or rescuing princesses, the story is about the emotional entanglements of human relationships, specifically about the dissolution of a marriage. There is unity of time and space – all action takes place in an apartment – and the overall event structure is modulated to align to a well-formed Aristotelian tension arc, i.e. inciting incident, rising tension, crisis, climax, and denouement, independent of the details of exactly what events occur in any one run-through of the experience.

Additionally, the story-level choices in *Façade* are intended to not feel like obvious branch points. We believe that when a player is faced with obvious choice points consisting of a small number of choices (for example, being given a menu of three different things to say to choose from), it detracts from the sense of agency; the player feels railroaded into doing what the designer has dictated. Instead, in *Façade*, the story progression changes in response to many small actions performed by the player throughout the experience.

Section 4 of this paper describes *Façade*'s procedural content in detail, and how it achieves these design goals.

3.2 Hindrances Of Low- or Non-Procedural Content

Authors have faced a long-time conundrum when undertaking the construction of interactive stories: how can a story be structured to incorporate interaction, yet retain a satisfying, well-formed plot when experienced by the reader/player? Historically the designs of low- or non-procedural interactive stories have been forced to make a tradeoff between these two goals. The resulting "interactive story" may have a well-formed plot, but can only be minimally influenced by the reader/player, as seen in the linear narrative threads of most games and some text-adventure interactive fiction (IF).

Alternatively, the design tradeoff may be made in the other direction, resulting in interactive experiences that can vary significantly as a result of player action, but lacking the degree of coherency, pacing and focus that is pleasurable about well-constructed stories. A non-procedural, encyclopedic design approach in which the author creates a (large) number of static story pieces (assets) that are sequenced by a simple system, inevitably forces this design tradeoff. The author can choose to place minimal constraints on the ordering of story pieces, allowing the local sequencing of pieces to depend on the local player interaction. But then the sequences produced will lack the coherency of well-formed story arcs. Fragmented plots, or plots

heavily diluted with unorganized or non-useful bits of action, are common in hypertext fiction as well as some IF, making them problematic to characterize as proper stories.

The only way to increase interactivity in an encyclopedic design approach is to author extraordinary amounts of content by brute force. Even the most successful *Choose Your Own Adventure* books, where the plot may vary significantly in response to reader's choices *and* be well-formed, necessarily offer an unsatisfyingly short series of infrequent, binary choices in order to avoid a combinatorial explosion of explicitly rendered (pre-written) plot directions. Thus the limited and cumbersome nature of a non-procedural, encyclopedic approach is exposed.

Based on frustrating limitations in the prior approaches described above, local and global agency within interactive stories have commonly been seen as incompatible.

3.3 Procedural Story Design

Our solution in *Façade* to this long-time conundrum is to recast player interactions within a story in terms of abstract *social games*. Games, which are procedural by nature, achieve the high degree of event variability and player agency that we desire; the challenge becomes how to design and structure games that reflect the particular meanings we wish our story to exhibit, and how to *dramatically perform* the games as coherent, focused, well-paced narratives.

Further, to be compatible with the procedural, simulation-oriented nature of games, the granularity of immutable story content pieces must be made unusually small, on the order of individual and re-combinable facial expressions, gestures and lines of dialog, rather than multi-sentence lexias of text or extended cutscenes. As described in detail in Section 4, *Façade*'s content pieces are organized into multiple, mixable hierarchical levels, sequenced by procedures written in multiple, mixable authoring languages.

At a high level, *Façade's* abstract social games are organized around a numeric "score", such as the affinity between a character and the player. However, unlike traditional video games where there is a fairly direct connection between player interaction (e.g. pushing a button to fire a gun) and score state (e.g. a decrease in the health of a monster), *Façade's* social games have several levels of abstraction separating atomic player interactions from changes in social "score". Instead of jumping over obstacles or firing a gun, in *Façade* players fire off a variety of *discourse acts* in natural language, such as praise, criticism, flirtation and provocation. While these discourse acts will generate immediate reactions from the characters, it may take story-context-specific patterns of discourse acts to influence the social game score. Furthermore, the score is not directly communicated to the player via numbers or sliders, but rather via enriched, theatrically dramatic performance.

As a friend invited over for drinks at a make-or-break moment in the collapsing marriage of the protagonists Grace and Trip, the player unwittingly becomes an antagonist of sorts, forced by Grace and Trip into playing psychological "head games" with them [3]. During the first part of the story, Grace and Trip interpret all of the player's discourse acts in terms of a zero-sum *affinity game* that determines whose side Trip and Grace currently believe the player to be on. Simultaneously, the *hot-button game* is occurring, in which the player can trigger incendiary topics such as sex or divorce, progressing through tiers to gain more character and backstory information, and if pushed too far on a topic, affinity reversals. The second part of the story is organized around the *therapy game*, where the player is (purposefully or not) potentially increasing each characters' degree of self-realization about their own problems, represented internally as a series of counters. Additionally, the system keeps track of the overall story *tension level*, which is affected by player moves in the various social games. Every change in each game's state is performed by Grace and Trip in emotionally expressive, dramatic ways. On the whole, because their attitudes, levels of self-awareness, and overall tension are regularly progressing, the experience takes on the form and aesthetic of a loosely-plotted domestic drama.

As the granularity of the atomic pieces of story content (e.g. dialog, emotion and gestural expression) becomes very small, and the procedures to sequence and combine them into a coherent narrative performance become primary to the realization of the experience for the player, the author's activity shifts from that of a writer of prose into a writer of procedures, that is, into becoming a programmer.

4. PROCEDURAL CONTENT IN *FAÇADE*

4.1 Richness Through Coherent Intermixing

To dramatically perform *Façade*'s social games (introduced in Section 3.3) as coherent, focused, well-paced narratives, an organizing principle is required that breaks away from the constraints of traditional branching narrative structures, to avoid the combinatorial explosion that occurs with complex causal event chains [5]. Our approach to this in *Façade* is twofold: first, we divide the narrative into *multiple fronts of progression*, often causally independent, only occasionally interdependent. Second, we build a variety of *narrative sequencers* to sequence these multiple narrative progressions. These procedural sequencers, described below, operate in parallel and can coherently intermix their performances with one another.

Façade's architecture and content structure are two sides of the same coin, and will be described in tandem; along the way we will describe how the coherent intermixing is achieved.

4.1.1 Architecture and Content Framework

The *Façade* system consists of several procedural subsystems that operate simultaneously and communicate with one another [11, 12, 13, 14, 15]. Each is briefly described below.

The dynamic, moment-by-moment performance of the characters Grace and Trip – how they perform their dialog, how they express emotion, how they follow the player around and use objects – are written as a vast collection of *behaviors*, which are short reactive procedures representing numerous goals and sub-goals for the characters, arranged in a vast, hierarchical, dynamically-changing tree structure. These behaviors are written in a reactive-planning language called A Behavior Language (ABL), developed as part of the *Façade* project, that manages both parallel and sequential behavior interrelations such as sub-goal success and failure, priority, conflict, preconditions and context conditions.

The narrative sequencers for the social games are also written in ABL, taking advantage of ABL's ability to perform *meta-*

behaviors that modify the runtime state of other behaviors; more on this in the next section.

The highest level narrative sequencer, a subsystem called the *drama manager*, sequences dramatic *beats* according to specifications written in a custom drama management language. Beats in *Façade* are large groups of behaviors organized around a particular topic, described in the next section.

Another subsystem is a set of rules for understanding and interpreting natural language (NL) and gestural input from the player. These rules are written in a custom language implemented with Jess, a forward-chaining rule language. When the player enters dialog, these NL rules interpret one or more meanings (the aforementioned *discourse acts*). A second set of rules called *reaction proposers* further interpret these DA's in context-specific ways, such as agreement, disagreement, alliance, or provocation, and send this interpretation to the behaviors and drama manager to react to.

The final subsystem is a custom animation engine that performs character action, emotional expression and spoken dialog by way of real-time non-photorealistic procedural rendering, as well as music and sound. The animation engine is driven by the ABL behaviors; the engine also senses information about the location and actions of each character for the behaviors to use.

4.1.2 Beats, Beat Goals and Beat Mix-ins

Façade's primary narrative sequencing occurs within a beat, inspired by the smallest unit of dramatic action in the theory of dramatic writing [16]; however *Façade* beats ended up being larger structures than the canonical beats of dramatic writing. A *Façade* beat is comprised of anywhere from 10 to 100 *joint dialog behaviors* (JDBs), written in ABL. Each beat itself is a narrative sequencer, responsible for sequencing a subset of its JDBs in response to player interaction. Only one beat is active at any time. A JDB, *Façade*'s atomic unit of dramatic action (and closer to the canonical beat of dramatic writing) consists of a tightly coordinated, dramatic exchange of 1 to 5 lines of dialog between Grace and Trip, typically lasting a few seconds. JDBs consist of 40 to 200 lines of ABL code. A beat's JDBs are organized around a common narrative goal, such as a brief conflict about a topic, like Grace's obsession with redecorating, or the revelation of an important secret, like Trip's attempt to force Grace to enjoy their second honeymoon in Italy. Each JDB is capable of changing one or more values of story state, such as the affinity game's value, or any of the therapy game's self-revelation progression counters, or the overall story tension level. Within-beat narrative sequencers implements the affinity game; the topic of a beat is a particular instance of the affinity game.

There are two typical uses of JDBs within beats: as *beat goals* and *beat mix-ins*. A beat consists of a canonical sequence of narrative goals called beat goals. The typical canonical sequence consists of a transition-in goal that provides a narrative transition into the beat (e.g. bringing up a new topic, perhaps connecting it to the previous topic), several body goals that accomplish the beat (in affinity game beats, the body goals establish topic-specific conflicts between Grace and Trip that force the player to choose sides), a wait goal in which Grace and Trip wait for the player to respond to the head game established by the beat, and a default transition-out that transitions out of the beat in the event of no player interaction. In general, transition-out goals both reveal information and communicate how the player's action within the beat has changed the affinity dynamic.

A beat's canonical beat goal sequence captures how the beat would play out in the absence of interaction. In addition to the beat goals, there are a set of meta-behaviors called *handlers* that wait for specific interpretations of player dialog (discourse acts), and modify the canonical sequence in response, typically using beat mix-ins. That is, the handler logic implements the custom narrative sequencer for the beat. Beat mix-in JDBs are beat-specific reactions used to respond to player actions and connect the interaction back to the canonical sequence. Handlers are responsible both for potentially adding, removing and re-ordering future beat goals, as well as interjecting beat mix-ins into the canonical sequence. By factoring the narrative sequencing logic and the beat goals in this way, we avoid having to manually unwind the sequencing logic into the beat goal JDBs themselves.

For *Façade*, an experience that lasts ~20 minutes and requires several replays to see all of the content available (any one runthrough performs at most 25% of the total content available), we authored ~2500 JDBs. Approximately 66% of those 2500 are in beat goals and beat mix-ins, organized into 27 distinct beats, of which ~15 are encountered by the player in any one runthrough (see the drama management section further below).

4.1.3 Global Mix-in Progressions

Another type of narrative sequencer, that operates in parallel to and can intermix with beat goals and beat mix-ins, are *global mix-ins*. (How coherent intermixing is achieved is described in a later section.) Each category of global mix-in has three tiers, progressively digging deeper into a topic; advancement of tiers is caused by player interaction, such as referring to the topic. Each tier in the progression is constructed from one or more JDBs, just like beat goals or beat mix-ins. They are focused on satellite topics such as marriage, divorce, sex, therapy, or about objects such as the furniture, drinks, their wedding photo, the brass bull, or the view, or as generic reactions to praise, criticism, flirtations, oppositions and the like. Additionally, there are a variety of generic deflection and recovery global mix-ins for responding to overly confusing or inappropriate input from the player. In total there are ~20 instances of this type of narrative sequencer in *Façade*, comprising about 33% of the total ~2500 JDBs.

4.1.4 Drama Management (Beat Sequencing)

The coarsest narrative sequencing in *Façade* occurs in the drama manager, or *beat sequencer*. This lies dormant most of the time, only active when the current beat is finished or is aborted (by the beat's own decision, or by a global mix-in). It is at the beat sequencing level where causal dependence between major events is handled – that is, where high-level plot decisions are made.

In a beat sequencing language the author annotates each beat with selection knowledge consisting of preconditions, weights, weight tests, priorities, priority tests, and story value effects – the overall tension level, in *Façade*'s case. Given a collection of beats represented in the beat language, such as the 27 listed in Table 1, the beat sequencer selects the next beat to be performed. The unused beat whose preconditions are satisfied and whose story tension effects most closely match the near-term trajectory of an

author-specified story tension arc (in *Façade*, an Aristotelian tension arc) is the one chosen; weights and priorities also influence the decision. [13].

PlayerArrives, TripGreetsPlayer, PlayerEntersTripGetsGrace, GraceGreetsPlayer, ArgueOverRedecorating, ExplainDating-Anniversary, ArgueOverItalyVacation, FixDrinksArgument, PhoneCallFromParents, TransitionToTension2, GraceStorms-ToKitchen, PlayerFollowsGraceToKitchen, GraceReturns-FromKitchen, TripStormsToKitchen, PlayerFollowsTripTo-Kitchen, TripReturnsFromKitchen, TripReenactsProposal, BlowupCrisis, PostCrisis, TherapyGame, Revelations-Buildup, Revelations, EndingNoRevelations, EndingSelfRevelations-Only, EndingRelationshipRevelationsOnly, EndingBothNot-FullySelfAware, EndingBothSelfAware

Table 1. The names of *Façade*'s 27 beats.

Subsequent sections on Coherent Intermixing, and Failures and Successes, further discuss beat sequencing.

4.1.5 Long-term Autonomous Mix-in Behaviors

Long-term autonomous behaviors, such as fixing drinks and sipping them over time, or compulsively playing with an advice ball toy, last longer than a 60-second beat or a 10-second global mix-in. While perhaps performing only a minor narrative function, occasionally mixing in a JDB into the current beat (comprising only 1% of *Façade*'s JDBs), they contribute a great deal to the appearance of intelligence in the characters, by having them perform extended, coherent series of low-level actions in the background over the course of many minutes, across several beat boundaries. By simultaneously performing completely autonomous behaviors and joint behaviors, *Façade* characters are a hybrid between the "one-mind" and "many-mind" extremes of approaches to agent coordination, becoming in effect "multi-mind" agents [11].

4.2 Strategies for Coherent Intermixing

Since global mix-ins for the hot-button game are sequenced among beat goals/mix-ins for the affinity game, which both operate in parallel with the drama manager that is occasionally progressing overall story tension, several strategies are needed to maintain coherency, both in terms of discourse management and narrative flow.

First, global mix-in progressions are written to be causally independent of any beats' narrative flow. For example, while quibbling about their second honeymoon in Italy, or arguing about what type of drinks Trip should serve (affinity game beats, chosen by the drama manager), it is safe to mix in dialog about, for example, sex, or the wedding photo (hot-button game mix-ins, triggered by a player's reference to their topics). Each mix-in's dialog is written and voice-acted as if they are slightly tangential topics that are being jutted into the flow of conversation ("Oh, that photo, yeah, it's really...").

At the discourse level, mechanisms exist for smoothly handling such interruptions. During a beat goal, such as Trip's reminiscing about the food in Italy, if a global mix-in is triggered, such as the player picking up (thereby referring to) the brass bull, a gift from Trip's lover, the current Italy beat goal will immediately stop mid-performance, and the brass bull global mix-in will begin performing, at whichever tier that hot-button game has already progressed to. At the time of interruption, if the Italy beat goal had not yet passed its *gist point*, which is an author-determined point in a beat goal's JDBs, it will need to be repeated when the global mix-in completes. Short alternate uninterruptible dialog is authored for each beat goal for that purpose. Also, each beat goal has a *reestablish* JDB that gets performed if returning to the beat from a global mix-in ("So, I was going to say, about Italy..."). Mix-in's themselves can be interrupted by other mix-in's, but if so, are not repeated as beat goals are.

With only a few exceptions, the narratives of affinity game beats themselves are also designed to be causally independent of one another. For example, in terms of maintaining coherency, it does not matter which order Grace and Trip argue about Italy, their parents, redecorating, fixing drinks, or their dating anniversary. When beat sequencing, this allows the drama manager to prefer sequencing any beats related to past topics brought up by the player. Likewise, hot-button mix-ins can be safely triggered in any order, into almost any beat at any time.

However, great authorial effort was taken to make the *tone* of each beat goal/mix-in and global mix-in match each other during performance. Most JDBs are authored with 3 to 5 alternates for expressing its narrative content at different combinations of player affinity and tension level. These include variations in word choice, voice-acting, emotion, gesture, and appropriate variation of information revealed. By having the tone of hot-button global mix-ins and affinity game beat goals/mix-ins always match each other, players often perceive them as causally related, even though they are not. Additionally, for any one tone, most JDBs are authored with 2 to 4 dialog alternates, equivalent in narrative functionality but helping create a sense of freshness and non-roboticness in the characters between runthroughs of the drama.

4.3 Characterizing Agency in *Façade*

In this section we attempt to characterize the resulting degree of local and global agency achieved in *Façade.* Creating player agency was a primary design goal for *Façade*, afforded by our approach of authoring highly procedural content.

4.3.1 Local Agency

When the player's actions cause immediate, context-specific, meaningful reactions from the system, we call this *local agency*. Furthermore, the greater the range of actions the player can take, that is, the more expressive the interface, then the richer the local agency (again, if the responses are meaningful).

Façade offers players a continuous, open-ended natural language interface, as well as physical actions and gestures such as navigation, picking up objects, hugging and kissing. The millions of potential player inputs are mapped, using hundreds of aforementioned NL rules, into one or more of ~30 parameterized discourse acts (DA's) such as praise, exclamation, topic references, and explanations; a second set of rules called reaction

proposers interpret these DA's in context-specific ways, such as agreement, disagreement, alliance, or provocation.

Ideally there would be immediate, meaningful, context-specific responses available at all times for all DA's. In the actual implementation of *Façade*, in our estimation this ideal is reached ~25% of the time, where the player has a satisfying degree of real-time control over Grace and Trip's emotional state, affinity to the player, which topic is being debated, what information is being revealed, and the current tension level. But more often, ~40% of the time, only a partial ideal is reached: the mapping/interpretation from DA to reaction is coarser, the responses are more generic and/or not as immediate. Furthermore, ~25% of the time even shallower reactivity occurs, and ~10% of the time there is little or no reactivity. These varying levels of local agency are sometimes grouped together in temporal clusters, but also have the potential to shift on a moment-by-moment basis.

There are two main reasons for these varying levels of local agency. First, from a design perspective, at certain points in the overall experience it becomes necessary to funnel the potential directions of the narrative in authorially preferred directions, to ensure dramatic pacing and progress. Second, and more often the case, a lack of local agency is due to limitations in how much narrative content was authored (see the Failures section below).

4.3.2 Global Agency

The player has *global agency* when the global shape of the experience is determined by player action. In *Façade* this would mean that the final ending of the story, and the particulars of the narrative arc that lead to that ending, are determined in a smooth and continuous fashion by what the player does, and that at the end of the experience the player can understand how her actions led to this storyline.

Façade attempts to achieve global agency in a few ways. First, beat sequencing (i.e., high level plot) can be influenced by what topics the player refers to; the sequencing can vary within the number of allowed permutations of the beats' preconditions and tension-arc-matching requirements. Even with only 27 beats in the system, technically there are thousands of different beat sequences possible; however, since most beats are causally independent, the number of *meaningfully different* beat sequences are few.

More significant than variations of beat sequences ("what" happened) are variations within beats and global mix-in progressions ("how" it happened). A variety of patterns and dynamics are possible within the affinity, hot-button and therapy games over the course of the experience; in fact these patterns are monitored by the system and remarked upon in dramatic recapitulations in the BlowupCrisis beat halfway through the drama, and in the RevelationsBuildup beat at the climax of the drama. A calculus of the final "scores" of the various social games is used to determine which of five ending beats gets sequenced, ranging from either Grace or Trip revealing one or more big hidden secrets and then deciding to break up and leave, or both of them too afraid to do anything, or both them realizing so much about themselves and each other that they decide to stay together.

4.4 Failures and Successes of *Façade*

In this section we attempt to evaluate our results in creating the interactive drama *Façade*, whose design goals were strongly shaped by our procedural content-centric approach to implementation.

4.4.1 Agency

During the production of *Façade*, within our "limited" authoring effort (beyond the building of the architecture, *Façade* required ~3 person-years of just authoring, which is more than a typical art/research project but far less than a typical game industry project) we made the tradeoff to support a significant degree of local agency, which came at the expense of not supporting as much global agency. Combined with the reality that the time required to design and author JDBs is substantial, only 27 beats were created in the end, resulting in far lower global agency than we initially hoped for. As a result, we feel we did not take full advantage of the power of the drama manager's capabilities.

Furthermore, because the specification of each joint dialog behavior – spoken dialog, staging directions, emotion and gesture performance – requires a great deal of authoring and is not automatically generated by higher-level behaviors or authoring tools, we are limited to the permutations of hand-authored, intermixable content. *Façade* is not generating sentences themselves – although it is generating sequences.

4.4.2 Feedback

A major challenge we encountered, that we believe *Façade* falls short on, is always clearly communicating the state of the social games to the player. With traditional games, it is straightforward to tell players the game state: display a numeric score, or show the character physically at a higher platform, or display the current arrangement of game pieces. But when the "game" is ostensibly happening inside of the characters' heads, and if we intend to maintain a theatrical, performative aesthetic (and not display internal feelings via stats and slider bars, ala *The Sims*), it becomes a significant challenge. In our estimation *Façade* succeeds better at communicating the state of the simpler affinity and hot-button games than the more complex therapy game.

4.4.3 Interface

Another major challenge was managing the player's expectations, raised by the existence of an open-ended natural language interface. We anticipated natural language understanding failures, which in informal evaluations of *Façade* to date, occur ~30% of the time on average. This tradeoff was intentional, since we wanted to better understand the new pleasures that natural language can offer when it *succeeds*, which in *Façade* we found occurs ~70% of the time, either partially or fully.

4.4.4 System Architecture

In our estimation, a success of *Façade* is the integration of the beat goal/mix-in, global mix-in and drama manager narrative sequencers, with an expressive natural language interface, context-specific natural language processing, and expressive real-time rendered character animation. We feel the overall effect makes some progress towards our original design goals of creating a sense of the immediacy, presence, and aliveness in the characters required for theatrical drama.

4.4.5 Design

Certain aspects of our drama's design help make *Façade* a pleasurable interactive experience, while others hurt. It helps to have *two* tightly-coordinated non-player characters who can believably keep dramatic action happening, in the event that the player stops interacting or acts uncooperatively. In fact, the fast pace of Grace and Trip's dialog performance discourages lengthy natural language inputs from the player. By design, Grace and Trip are *self-absorbed,* allowing them to occasionally believably ignore unrecognized or unhandleable player actions. Creating a loose, sparsely plotted story afforded greater local agency, but provided fewer opportunities for global agency. However, the richness of content variation, and at least moderate degree of global agency achieved, does encourage replay.

The huge domain of the drama, a marriage falling apart, arguably hurt the success of the overall experience, in that it overly raised players' expectations of the characters' intelligence, psychological complexity, and language competence. As expected, the system cannot understand, nor has authored reactions for, many reasonable player utterances. The large domain often requires mapping millions of potential surface texts to just a few discourse acts, which can feel muddy or overly coarse to the player. Also, continuous real-time interaction, versus discrete (turn-taking) and/or non-real-time interaction, added a great deal of additional complexity and authoring burden.

5. CONCLUSION

In this paper we have argued that procedural authorship is required to take full advantage of the representational power of the computer as an expressive medium. Procedurality is an underlying support for all modes of digital authorship; while procedural literacy is not required to create digital work, new media practitioners without procedural literacy are confined to producing those interactive works that happen to be possible to produce within existing authoring tools. We attempted to make a case for the importance of procedural authorship, describing the design goals of a case-study, the interactive drama *Façade,* and how these goals could only be met through a highly procedural approach to interactive narrative. *Façade's* architecture, content organization, and the experience of authoring within this framework were described in detail and evaluated, determining that procedural authoring is an essential method for enabling yet-to-be-realized genres of interactive art and entertainment.

6. REFERENCES

[1] Albee, E. 1962. *Who's Afraid of Virginia Woolf?* Signet.

[2] Bates, J. 1992. "Virtual Reality, Art, and Entertainment." *Presence: The Journal of Teleoperators and Virtual Environments* 1(1): 133-138.

[3] Berne, E. 1964. *Games People Play*. New York: Grove Press.

[4] Crawford, C. 1987. "Process Intensity". *The Journal of Computer Game Development*, Vol. 1, No. 5. http://www.erasmatazz.com/library/JCGD_Volume_1/Process_Intensity.html.

[5] Crawford, C. 1989. "Indirection". *The Journal of Computer Game Design, Volume 3*. http://www.erasmatazz.com/library/JCGD_Volume_3/JCGD_Volume_3_Index.html

[6] Grand Text Auto. 2005. "Finally, the Curtain Opens on *Façade*". http://grandtextauto.gatech.edu/2005/07/05/facade-is-released/

[7] InteractiveStory.net. 2005. *Façade, a one-act interactive drama*. http://www.interactivestory.net

[8] Laurel, B. 1991. *Computers as Theatre*. Reading, MA: Addison-Wesley.

[9] Loyall, A. B. 1997. *Believable Agents.* Ph.D. thesis, Tech report CMU-CS-97-123, Carnegie Mellon University.

[10] McCorduck, P. 1991. *Aaron's Code: Meta-art, Artificial* Intelligence*, and the Work of Harold Cohen*. New York, NY: W. H. Freeman and Co.

[11] Mateas, M. and Stern, A. "A Behavior Language: Joint Action and Behavioral Idioms", in Predinger, H. and Ishiuka, M. (eds.) *Life-like Characters: Tools, Affective Functions and Applications*, Springer 2004.

[12] Mateas, M. and Stern, A. "Natural Language Understanding in *Façade*: Surface Text Processing", in Proceedings *of Technologies for Interactive Digital Storytelling and Entertainment*, Darmstadt, Germany, 2004.

[13] Mateas, M. and Stern, A. "Integrating plot, character and natural language processing in the interactive drama *Façade*." In *1st International Conference on Technologies for Interactive Digital Storytelling and Entertainment (TIDSE '03)*, Darmstadt, Germany , March 24 – 26, 2003 .

[14] Mateas, M. and Stern, A. 2003b. "*Façade*: an experiment in building a fully-realized interactive drama". *Game Developers Conference (GDC '03)*, San Jose, CA, USA, March 4 – 8, 2003.

[15] Mateas, M. and Stern, A. 2000. "Towards Integrating Plot and Character for Interactive Drama." In *Working notes of the Social Intelligent Agents: The Human in the Loop Symposium*. AAAI Fall Symposium Series. Menlo Park, CA.: AAAI Press.

[16] McKee, R. 1997. *Story: Substance, Structure, Style, and the Principles of Screenwriting.* New York: HarperCollins.

[17] Murray, J. 1998. *Hamlet on the Holodeck.* Cambridge, MA: MIT Press.

[18] Wikipedia, 2005. http://en.wikipedia.org/wiki/Demoscene