MAST (Metaphoric Adventure-Scripting Tool): A user facing tool for creating multiplayer MMO narratives

Committee: Dr. Celia Pearce (chair), Dr. Janet Murray, Dr. Ian Bogost
Contents

Abstract ......................................................................................................................................................... 4

Chapter 1: Introduction ................................................................................................................................ 6
  Context ...................................................................................................................................................... 8
  Goals ....................................................................................................................................................... 11

Chapter 2: Design Theory ............................................................................................................................ 14
  The MMO Quest .......................................................................................................................................... 14
    As Unit of Interaction .............................................................................................................................. 19
    As 'Environment' in a Virtual World ........................................................................................................ 21
    As Storytelling Device ............................................................................................................................. 25
      The Myth of Player Choice .................................................................................................................. 26
    Cinematics and Film Envy ....................................................................................................................... 31
  User-Created Content ................................................................................................................................. 36
    Comparison Studies ................................................................................................................................ 38
      Mission Architect ................................................................................................................................ 38
      Spore: Galactic Adventures .................................................................................................................. 40
      Electron Toolset (Neverwinter Nights 2) ............................................................................................ 41
    Conclusions ............................................................................................................................................. 43

Chapter 3: Artifact Design ........................................................................................................................... 45
  Introduction to MAST ............................................................................................................................... 45
    Design Process .................................................................................................................................... 46
    Design Philosophy ............................................................................................................................... 48
    Example User Story ............................................................................................................................. 50
  Design Process ........................................................................................................................................ 51
  Architectural Design ................................................................................................................................ 53
  Final Design Iteration for MAST .............................................................................................................. 56
  GM and Player Tools Design .................................................................................................................... 67
    Features .............................................................................................................................................. 68
    Final Design Iteration for the GM Tools .............................................................................................. 74
    Final Design Iteration for the Player Tools .......................................................................................... 79
  Conclusion ............................................................................................................................................... 80
Abstract

For my thesis project, I examined how players interact with MMOG user development tools (excluding MMOWs/MMOVWs) to produce their own content. While MMOG content has been discussed as a) a site for discussing the cultural implications of mechanics\(^1\), b) categorizing MMOG players\(^2\), and c) as part of larger arguments about the state of game narrative\(^3\); very little attention has been paid to how players of these games interact with content on their own terms, and how (when given the opportunity) they might design content of their own\(^4\). As an MMOG player, but only an indifferent and infrequent MMOG role-player, I became interested in the idea of re-applying the techniques of table-top role-playing to their newest generation of digital counterparts. Out of the many directions I could approach this broader topic, I am most interested in extracting a preliminary set of 'best practices' and attempting to demonstrate an application of those practices in a tool designed for 5 Boroughs (the MMOG I have been designing through the {egg}). I'm approaching this work in the context of Pearce's Communities of Play, Fine's Shared Fantasy: Role Playing Games as Social Worlds, and Mackay's The Fantasy Roleplaying Game. Tabletop role-playing games produce a unique community dynamic, which, if encouraged by the availability of digital tools, would migrate to role-play in digital environments. At the broadest level, my hypothesis is that, given a tool for creating and sharing their own narrative content, players will choose to supplement and enhance their experiences with designer-produced content with content of their own; with demonstrable effects on player movement through a finite game space. More specifically, that it is possible to extract a formal system of storytelling which is sufficiently abstract to allow for many 'types' of story, while grounding the produced stories in the context of the world they are being

---

MAST (Metaphoric Adventure-Scripting Tool): A user facing tool for creating multiplayer MMO narratives designed for; and, with it, foster a co-narrative relationship between the development and player community. To that end, I will develop both a set of design principles for developing user content tools, and a sample tool that follows those principles, informing my design with academic research, and industry experience. Methodologically, after establishing my set of principles, I will iteratively test paper prototypes of that design, and begin implementing its key functionality. In essence, the primary question of this project is whether a generalize-able set of user content tool 'rules' can be developed which are both usefully descriptive (allowing a designer to rule out directions of development), genre-unspecific (applicable to a variety of styles of content-driven game), and provide a useful context in which to consider user 'play flow' (helping the tool to become part of normal play for most players).
Chapter 1: Introduction

Massively multiplayer online games pose a unique design challenge: how do you provide a large, content-rich environment for an inherently migratory population? Traditional MMORPG mechanics rely on progression and escalation (kill monsters to level up to kill bigger monsters, ad infinitum). To keep up with that treadmill of task/reward/bigger task, players must constantly move through game spaces as itinerant populations, looking for the next level-appropriate locus. This leads to two significant problems in online game design: community maintenance and development bottlenecks. MMORPGs thrive on persistent communities, players who continue to traverse the game space (whether for mastery, socialization, or completionism), and communities emerge most naturally where there is a visible player presence throughout the world. These visible players are attachment points for new players entering a game without a pre-existing social network, and are necessary for the retention of said players. Since players are mechanically discouraged from 'settling' or really inhabiting specific game spaces, most MMORPGs (outside of level-cap areas) feel curiously empty. Game content production is more resource-intensive to produce than it is to consume. And because players are constantly looking for new tasks (affording chances to collect experience or money), they are constantly moving to new areas, consuming the existing content too quickly for content development to keep up, leading to endemic content deficits.

Because content is consumed so quickly, faster than it can be created; designers fall back on 'cookie cutter' quests, which encourage a mindless approach to completing them (frequently called 'grinding'). What little narrative is attached to a quest is merely a wall between the player and the MacGuffin⁶, or the next thing they need to kill fifteen of; so it's hardly a mystery when most people don't care about

---

⁶ "A plot element that catches the viewers' attention or drives the plot of a work of fiction"; MacGuffin, Princeton University, WordNet 3.0
that narrative or the context within which it exists. And more than simply being boring for players to complete, they're boring for designers to create (see the gradual decline of player-produced content using The *City of Heroes* Mission Architect for ). Most tools designed explicitly for player-created content are either too constrained (*The City of Heroes* Mission Architect) or too open-ended (*Second Life*), so how can we balance these points? How do we create a tool that allows creative freedom within a preexisting story world and its canon? The solution presented in this thesis solves this problem by borrowing from two well-established practices, Tarot card storytelling and GMing, to create a more game-like experience (which is itself enjoyable), gives players some guidance while also allowing them a significant amount of creative freedom, and allowing the GM to continue participating in the created narrative (as they would in a tabletop game).

Essentially, my project connects industry-level issues in content development and community building with persistent academic concerns about the narrative agency of players in online worlds. I am proposing that collaborative (or 'co' narration) is a tool that can be applied to addressing this issue. To create the circumstances which support co-narration, conditions must be met for both players and developers. Players must be able to both create content that exists in real game spaces (however access might be moderated), and receive feedback on that content. Developers must be able to moderate and rate player content, and be able to manage official or 'canonical' content channels. Without having creative influence (if not control), few developers will be willing to allow player-created materials into their 'themepark' space. Without creative investment and responsive narrating partners, few players will be willing to produce content according to a seemingly arbitrary set of story-telling 'rules'. By allowing players to make 'real' content, but restricting their ability to make 'official' content, developers can both allow players to use their game spaces more effectively, and establish a closer narrative relationship with storytellers operating within the established tone and character of the world.
Co-narration is a fairly tall order for a single development tool offering broadly defined creative freedoms; much less a tool which bridges the gap between players and developers in any MMORPG.

Fortunately, MAST will be prototyped within an MMOG uniquely suited to encouraging a relationship between developers and a player community: 5 Boroughs.

**Context**

For nearly two years, as part of my research within the Emergent Game Group (egg), I have been engaged in two tasks: prototyping and evaluating strategies for scaffolding emergent behavior in online games, and exploring how MMOGs can be an effective method for engaging with historical knowledge and in historical practices; providing unique affordances for learning. 5 Boroughs is the result of that research; a (nearly) fully-designed MMOG currently being prototyped by a mixed team of graduate and undergraduate students.

5 Boroughs is a real-time, persistent, narrative MMOG, which draws from both from the traditions of exploratory social games like Myst Online: Uru Live and the strong narrative/mechanic relationship found in games like Dangerous High School Girls in Trouble. 5 Boroughs is set in New York City during the Ellis Island Era (1892-1954); and is designed to allow players to assume control of a virtual world populated primarily by other people. The central mechanic is card-based, with tarot cards standing in for 'life experience' used to solve narrative problems in the larger world.

Within this historical context, players have the freedom to direct not only the course of their own lives and destinies, but that of the world at large. They can contribute to the creative and intellectual life of their community by participating in oral traditions and folk activities, write and publish literary materials, produce art, get involved in social or political organizations (labor unions, religious groups, or organized crime, for example), or open their own businesses. In the process, they might work as
trailblazers to overcome cultural and economic barriers in an era of racial and sexual discrimination, or attempt to balance the construction of vibrant immigrant communities against the desire to assimilate into the American identity; all alongside (or in opposition to) historical figures of the day. In so doing, players actively engage with their own cultural history, and gain a nuanced understanding of the role immigration plays in that history.

By emphasizing social play at the narrative and mechanical level, 5 Boroughs examines the subjective experience of cultural assimilation; giving teens and adults the opportunity to diagnostically 're-create' modern New York City. Literal role-playing interaction of this kind can provide special and unique insights (through the application of subjectivity) into the factual information about a period. In attempting to design the mechanics through which immigrant characters (in Passage) would participate and initiate these group narrative experiences, it became clear that a separate tool (and interface) would need to be developed. This tool, would allow a player to manage storylines they were currently developing, maintain a record of completed and 'in-progress' narratives, and 'build' narratives for other players using some sort of scripting interface with a scalable level of access to programmatic complexity.

By virtue of our very specific and somewhat unusual design goals, 5 Boroughs is uniquely suited for testing an application like MAST. It is a primarily narrative game, emphasizing personal narratives contextualized within a larger historical narrative. And its central mechanic represents the collection of life experiences, utilized through storytelling-like behaviors. So our design self-selects for the community most likely to be interested in telling one another interactive stories, and it's underlying structure supports a story-like form of interaction. Prototyping MAST through 5 Boroughs allows me to harness players who already want to tell stories in online worlds, and give them the tools to do so.

Those players, presumptively, will be coming from the tradition of tabletop role-playing games: Game Masters (GMs), Dungeon Masters (DMs), and Storytellers; people who already tell long and short form
stories within existing smaller communities\textsuperscript{7}. While some percentage also engage Chat Role-play in MMOGs (the most prevalent form of online role-play); anecdotally, a greater percentage attempt to cobble together gaming environments using forums, chat clients, dicebots, and assorted image sharing tools in order to mimic the tabletop experience. This is primarily because MMOG role-play combines the worst flaws of chat only interfaces (no visual contact, ambiguous text communication, and wandering attention) with a static, uncontrollable environment. Essentially, MMOGs (massively multiplayer online games) like World of Warcraft, and MMOVWs (massively multiplayer online virtual worlds) like Second Life essentially provide a high fidelity chat client. World of Warcraft provides a very specific context and mechanics, but no meaningful tools to affect the play environment. Second Life provides environmental tools, but no mechanics or context. GMs rarely work from nothing, but rather construct a story based in an existing rule system or narrative context. To draw GMs more fully into digital role-playing, they need to have (even if it is within a limited scope) access to all three. I’ll be talking about the role of storytelling in more depth in the beginning of Chapter 2: Design Theory.

As mentioned above, the central mechanic of 5 Boroughs is tarot card based, representing the ‘literal’ life experience of player characters; a fairly singular metaphor that requires a bit more discussion.

Briefly, the Tarot is a deck of seventy-eight cards, originating in mid-15th century Western Europe, which was used to play trick-taking card games. Tarot decks began to be used as divinatory tools in the late 18th century\textsuperscript{6}, and that use has dominated popular knowledge ever since. Like ordinary playing cards, tarot decks have four suits (which correspond to the suits of conventional playing cards). Each of these suits has ten pip (counting) cards from ace to ten, and four face cards (minor arcana) for a total of fourteen cards. Tarot decks are distinguished by an additional twenty-two-card trump suit (major arcana). Each card has two or more meanings, based on its orientation (upright or reversed), it’s position


in a spread (in the future or the past), and the cards surrounding it. Broadly, the major arcana
(numbered zero to twenty-one) represent the evolution or maturation of an innocent, full of potential
(The Fool, card zero), to an adult who has achieved cosmic consciousness (The World, card twenty-
one)\(^9\). The minor arcana and the pip cards represent the ‘everyday’ trials and joys of life, and provide
guidance on the balance of life throughout ‘the Fool’s Journey’.

As a divinatory tool, tarot cards are shuffled and drawn randomly into 'spreads' of varying complexity;
from simple three-card spreads to the ten-card Celtic Cross spread or the Tree of Life spread. Those
cards are then interpreted, according to placement, card meaning, and card orientation. This
interpretation is shaped and defined by not only the mythic meanings of the cards, but the associations
of the interpreter, and their own life experiences. Interpreting a tarot card reading is frequently called
'reading', but might more accurately be called 'reading into'. This is represented most literally in Italo
Calvino’s *Castle of Crossed Destinies*, wherein the characters can only communicate their stories with
one another through the use of tarot cards\(^{10}\). In a very practical way, tarot decks are already used as a
storytelling tool, and while they will not necessarily be immediately familiar to any audience, the
symbolic meanings attached to them are easy to grasp and straightforward to teach. In *5 Boroughs* and
**MAST**, I am attempting to explicitly draw from this tradition of co-narration and group storytelling which
is built into the experience of using tarot cards. My proposed solution combines the well established
narrative form of tarot cards and GM traditions, to create a game-like narrative creation tool.

**Goals**

My goal with this project, then, is to prototype a narrative development tool that promotes a mutually
creative relationship between developers and players. To even approach a design goal so broad, my

---


primary theoretical question, and the question from which I derived most of my bibliography, was: how can we extract (through design gedankenexperiments and comparative game research) a formal system of co-narration? In traditional storytelling, a narrative is being created by one person (or persons) relating a series of events to another person or persons. In MMOGs the development team (the writers & designers of the game) are relating a series of events to the player(s) in the form of a narrative that relates some information. 'Responsiveness' then, is the degree to which the development team is listening to, and interacting with, their player co-narrators in that multiplayer world.\(^\text{11}\)

Next, I turned to the work of the \{egg\}. In an effort to build genuinely emergent narrative structures in MMOs, the \{egg\} has pursued several lines of research through development. Of these projects, I managed and developed 5 Boroughs, a heavily scaffolded\(^\text{12}\) environment aiming for a group narrative experience that is neither purely 'themepark' nor 'sandbox'. This gave me an environment to work within: bounding my search further by how co-narration could exist in this specific game. Because 5 Boroughs is a uniquely narrative-focused MMOG, which a heavy emphasis on problem-solving in a multiplayer space, I specifically chose to focus on narrative strategies (as opposed to more mechanical strategies to obviate the need for it) for addressing the content design issues raised above.

To that end, while I believe that these design principles are usefully universal, they were developed in a specifically chosen context, informed by my academic research (as well as the collective work of the \{egg\}), industry experience, supplemental ethnographic work, and design background with the 5 Boroughs/Passage project. The sample tool I have developed in accordance to those principles will then reflect this bias, which I will detail further in the following chapter. Chapter 2 will discuss the theory, practice, and literature that informed the design of MAST. Chapter 3 will include my research into other


artifacts with similar or related goals, discuss the role of User Generated Content in an MMOG, and step through the design of MAST, and the corresponding GM & Player Tools. This includes a screen by screen breakdown of the proposed artifacts, with notes detailing the functionality, workflow, and influences of that screen or process. Chapter 4 will demonstrate the process with an extensive walkthrough.
Chapter 2: Design Theory

This chapter will provide a background in the relationship between storytelling and play, and detail why 'questing' behavior is a useful lens for studying MMOGs. And how, from that groundwork, we can better understanding the relationship between 'quest' play and traditional role-playing; meaningfully affecting player investment in game content. I will be taking examples from general practice, narrative theory, the sociology of play, and personal experience; condensing them into an underlying philosophy which can be filtered through practical design principles; from which I will create a feature list for a hypothetical tool in Chapter 3.

The MMO Quest

First, broadly defining a few terms that will be used throughout the rest of document, I'm going to establish a few semantic differences between MMOGs (massively multiplayer online games) & MMOVWs (massively multiplayer online virtual worlds) in order to use these terms with more precision than an individual game's marketing identification would allow. While virtual worlds can encompass any form of online community, I will be using to describe the specific subgenre which takes place in a simulated visual environment, where users can interact with one another and use or create objects in that space. Ideologically, MMOVWs are intended for inhabitation and interaction-- an MMOVWs users frequently 'live' in specific areas, occasionally having explicitly purchased land to build interactive objects on. Online games have a backbone 'mechanic' that forms the basis of multiplayer interaction. An MMOVW can be an MMOG, and an MMOG can have MMOVW elements, but they do not wholly overlap.

Bishop, J. (2009). "Enhancing the understanding of genres of web-based communities: The role of the ecological cognition framework". International Journal of Web-Based Communities, 5(1), 4-17
As shown above, MMOGs in the colloquially understood sense (MMOVWs with a strong mechanical basis and limited player construction of objects) tend to be affiliated with top-down narrative content. It will have an established world narrative, which the player may consume at their leisure, which will persist in the world environment. Players can affect this story only insofar as they can choose a perspective to view that story from. MMOVWs in the colloquially understood sense (MMOVWs with significant visual fidelity and limited or no emphasis on multiplayer mechanics) tend to be affiliated with bottom-up narrative content. Unless it has been branded as affiliated with a specific other game (e.g. Uru: Myst Online) it will have no established world narrative, and players may establish enclaves of environmental narrative as they chose. Both ends of this spectrum allow for text chat and emoticon-based role-play, but none directly link the use of what mechanics they have to the creation of either ongoing world narrative or personal narratives. Along a separate axis we can compare 'themepark' and 'sandbox' games. Themeparks (as derived from their amusement park namesakes) are static, and strongly link narrative progression to a player's physical location in a world: go to the Hilsbrad Foothills for player content rated levels 20-30; go to The Flit when your Shadowy reaches at least 50; go to Adventureland from Cinderella's Castle. Sandboxes are dynamic, chaotic, and very weakly link narrative progression to physical location (and even then, typically only in starting areas like High Security Space.

Harðarson, Reynir. Themeparks and Sandboxes, IGDA Lecture. 2009
16 A route in Echo Bazaar (http://echobazaar.wikidot.com/flit).
in *EVE Online*). Finally, along a third axis, there are ‘fixed synthetic’ and ‘co-created’ worlds. Fixed synthetic worlds have a strong diagetic narrative which contextualizes player action, and foregrounds the creative vision of the designers. Co-narrative worlds have a no diagetic narrative to contextualizes player action, foregrounding the creative vision of the players. What we begin to see emerging here are three spectrums on which a given game can be evaluated: the mechanical underpinning of a world, who controls the 'lore' of that world (players or designers), and how much in-world 'content' is driven by player activity.

Within the context established above, MMOGs typically include both 'single-player' and 'multiplayer' content, and some allow what is called 'user-created' content. Single-player content, rather than identifying the number of players required to complete a given unit of narrative ('quest'), refers to the narrative of the world as experienced by a player's character. While a quest might require more than one player, it is held in singular; one player holds an instance of a quest which does not interact with another player's instance of the same quest. Both players can complete the objectives of that quest, return whatever deliverables it may have, and receive the same reward. The game narrative behaves like a single-player role-playing game (e.g. Oblivion, or the Final Fantasy series), wherein a single character is acting in the world, performing what tasks as they need to accomplish some eventual goal.

Multiplayer content, which may or may not require multiple players, refers to narrative which presupposes the existence of other agents (players) in the world, and which can be effected by the behavior of those other players. A significant portion of the 'end-game' content in *World of Warcraft*, for example, while requiring between five and forty players to complete, is single-player. There is no diagetic reference to other players engaging in this activity with you, and no non-mechanical significance to the fact that the encounter took place in a group rather than individually.

---

19 End Game Content: narrative and game play which requires max level and significant investment in high-level equipment to participate in.
So why is single player content problematic in MMOGs? It's clearly the dominant narrative paradigm, and it's successful enough as a player experience that that it is not preventatively bad as a narrative distribution strategy. To answer that question, we'll look a little more closely at what an MMOG Quest is, and why the currently accepted structure is a very poor fit for the narrative or environmental goals of quest, and only occasionally successful at meeting the goals associated with a 'unit of interaction'. This poor fit produces, if not a wholly unsuccessful user experience, a less than ideal one; one that cheapens player interaction with the world, reducing it to tourism; and one that is largely an uninspired time-sink for designers.

In literature, a quest is a dramatic narrative form in which a "...hero ...aims to obtain something or someone..., and with this object to return home." Here it bears noting that the 'quest' also plays a significant role in the Monomyth, which lends a lot to the genre of 'heroic fantasy' that far and away dominates the narratives of the MMOG industry. In fact, 'the quest' has become such a pervasive concept in defining narrative content that we have essentially loaded our terminology against being able to conceive of MMOG narrative content which is not fundamentally rooted in the monomythic form. And, on a trivial level, the quest of the Monomyth does mirror both the basic structure of an MMOG 'quest', and the player character's role in the MMOG world. The player advances through the world by taking on tasks, which require the collection and delivery of objects. And at their simplest, MMOG quests are a level design scavenger hunt, enlivened by the need to defend oneself against improbably irritable wildlife. Despite (or perhaps because of) that simplicity, quests are central to the user experience of an MMORPG. They teach game play. They provide scope (allowing players to frame a session of play) and direction. They are the primary form of interaction with the world. And most relevantly to our discussion here, they deliver the most atomic unit of narrative: a scene in a story,

---

punctuated by player action, within that context of 'epic' achievement and adventure, relying on the fundamental metaphor of player as hero.

But the technical and functional requirements that define a quest are just as constraining to the design process as the aesthetic: "From the designer’s point of view, a quest is a set of parameters in the game world (making use of the game’s rules and game play) that specifies the nature and order of events that make up a challenge for the player, including its resolution. From the player’s point of view, a quest is a set of specific instructions for action, they can be as vague as a general goal (overthrow the evil king) or extremely precise (take this bucket to the well, fill it up and bring it back to me); after the quest has been completed it can be narrated as a story." 2223

What a quest can do, mechanically, in a system, and what effects it needs to produce structurally or spatially, are just as important and defining as the plot (or narrative window dressing) of that story fragment. A quest serves many purposes in MMO content; and it needs to meet all of the requirements associated with those purposes. Requirements tend to broadly fall within the following categories:

- Extra-diagetic:
  - Does it teach game play?
  - Does it point players to a new area?

- Structural:
  - Where does it fit in a specific plot arc (a series of interlocking or related quest chains which flesh out some aspect of a game’s story world)?
  - Where does it fit in a zone arc (the complete story told about a part of the game world, as described in the quests that are used to fill that space)?

Where does it fit in a world arc (the over-arching narrative of a game; the story world as it unfolds around the player, traveling through that space)?

Or, in other words, how does it work to create the story world\textsuperscript{24} of a game?

- Diagetic: How does it involve the player in the narrative of the zone/world/etc? What story and character beats should it hit?

Practically speaking, quest design in MMOGs is meant to strike a careful balance between storytelling technique and structural constraint. So our most important content design (both in terms of designer developed and user created) considerations are player experience (extra-diagetic), world development (structural), and narrative engagement (diagetic). So we will analyze quests through these three lenses: as a unit of interaction, as an environmental supplement, and as a discrete unit of narrative.

**As Unit of Interaction**

'Game play'\textsuperscript{25}, or the *mechanical* interaction between game and player, as mediated by its rules, determines how a player can interact with an MMOG ("What can the player do?" "How does the player know when to do it?"). The rules associated with those mechanics (e.g. cool-downs, combinations, etc) provide the decision space and strategic complexity ("What could the player do now?"). But quests provide a context and a direction; they define the grades of challenge associated with subsets of interaction ("How difficult should I expect this quest to be?" "How valuable are its rewards?"). As a result, quests as a 'unit of interaction' (in western role-playing game derived MMOGs) serve several extra-diagetic purposes:

- Teach players about the mechanical underpinnings of the game world (game play)
- Direct players toward places where learned game-play can be used

\textsuperscript{25} Salen, Katie; Zimmerman, Eric (2004). Rules of Play: Game Design Fundamentals. Cambridge, Massachusetts: The MIT Press "Game play is the formalized interaction that occurs when players follow the rules of a game and experience its system though play."
• Allow players to scope progress within a zone, and to scope the duration of grouping (multiplayer activity).

• Direct players away from unavailable diatonic content or mechanics.

The design of an individual unit of interaction forms the abstract core of the player experience. A quest has a clear starting and ending point: accepting a quest from a NPC gives players a list of objectives, an end condition, and a projected reward. Abstractly, this should force designers to consider, at every step of the design process, if the player is performing the role in the story that is the most ‘fun’, according to the underlying mechanics of the system. Frequently, however, we see what looks like the opposite; quest design which de-emphasizes ‘the fun parts’ of a story and emphasizes the grinding or repetitive parts. The player spend the whole of their leveling experience collecting wolfbaby giblets for various scientists (or magicians) rather than graduating to more complicated tasks, like helping to perform the experiment (or ritual, etc). The current model of quest action/reward/follow-up leads to the following (semi) ridiculous scenario: the player collects 15 spare parts and brings them back to the goblin engineer who makes 5 bombs; the player then collects those bombs to blow up 12 dyspeptic alpacas; all as part of a larger arc which requires collecting 3 alpaca hides, brought to a shaman in an isolated part of a zone, who can use those hides to make the handle of the Lance of Longinus. All of the player action involves either killing a spawned creature, using an item to kill a creature, or ferrying inventory objects from one point of the map to another. And yet, this serves the four major tasks of the quest as a unit of interaction: it provides a game play option, directs the players to use it on a type of object, gives the player some idea of how long the task will take, and directs them away from what the mechanics of the game can’t do.

This is precisely the condition that MAST attempts to address: how to structure a creative tool that encourages players to create content which is ‘fun’ both to produce and to play; and also direct the
player to produce content which emphasizes game play and provides context for those game play options. While the mechanical system must take responsibility for the interactions it allows, the tools available to designers (or players) create the conceptual constraints seen above. Our tool design must make it both clear and fun to build content, allowing players to focus on creating, rather than learning the tool. In other words, we should prefer, and strive for in our player experience, breadth of exploration and the recombinatory 'joys of discovery' rather than a depth-focused analytical or 'deconstructible' tool. But our tool must also be conscious of the ways in which it will constrain and direct the content created in it. Essentially, since quests are the smallest unit of task performed in an MMO, tools for creating them should reflect both the literal mechanical restrictions of the underlying system and our least intrusive (and most diagetically sound) aesthetic for designing them.

**As 'Environment' in a Virtual World**

On a very basic level, the environment of an MMOG is created by linking together these units of interaction (creating quest flow), and spacing them within and between 'zones'. Quests create a sense of space, in part, by literally linking those spaces with goals, and creating inhabitants for those spaces (even if both the space and the inhabitants are static and unchanging). That spatial distribution of narrative allows us to think of game narrative in terms than the information it contains; and how the order in which quests are offered 'narrates' that space, contextualizing both its design, and its place in the larger story world.

---

26 'Quest flow' refers to the spacing of designer-created content in an MMOG, according to level and difficulty, to deliberately lead the player through each 'zone' of the world. (Mortensen, Torill. *Pleasures of the Player: Flow and Control in Online Games*. University of Bergen Doctoral Thesis, 2003.)

27 "Read in this light, a story is less a temporal structure than a body of information. The author of a film or a book has a high degree of control over when and if we receive specific bits of information, but a game designer can somewhat control the narraion process by distributing the information across the space." (Jenkins, Henry. "Game Design as Narrative Architecture." First person: New Media as Story, Performance, and Game. The MIT Press, 2004.)
A designed space should arguably speak for itself, but most MMO spaces cannot transition naturally, either narratively or visually. Quests explain, direct, and contextualize that space; covering over technical and spatial limitations. In other words, quests are structural as well as mechanical, and serve a highly defined set of goals:

- Fill a given space with discreet units of interaction
- Connect units of interaction into sequences of similar arcs (of greater or lesser complexity)
- Link spaces together in a specific (or parallel) order using these units of interaction
- Insert units of interaction which periodically spend a player back to major world locations to perform economic or mechanical tasks

By those requirements, 'grinding' is, if not good for level design, fairly neutral: it sends players back to a quest hub or city regularly, and provides units of interaction for the surrounding spaces. So why is grinding considered bad for game or interaction design (as distinct from narrative design, which will be discussed in the next section)? Grinding is, to some extent, a fact of life in MMORPG design at this stage. Players are already well trained in the model of 'perform X action to receive 1/100 Y'. So any quest which can be repeated for character rewards or specialization (i.e. through group affiliation) will result in players doing X until they have enough Y. And therein lays the problem; while grinding does not impede the structural function of a quest, it does make it harder for a quest to meet its goals as a unit of interaction; a quest which requires too many actions, or which is too challenging, is a quest which is too hard to repeat efficiently (necessary for grinding quickly). For efficiency to be possible, it must, to some extent be designed for; simple actions must be privileged over complex (reducing the complexity of actions available in designing any unit of interaction), and level or environmental design must assume

---

28 Grinding: repetitively performing the same task for a set reward of experience, reputation, or money.
29 See, for example Thrusting Hodir’s Spear, a quest performed in order to gain reputation with the Sons of Hodir, wherein you perform a rhythm game to wrestle a dragon.
the kinds of navigation necessary for grinding activities. This places requirements on the design of a space which break the continuity of the world, not narratively but visually and spatially; representing a breach of the quest ‘contract’, the myth that the player accomplished something by acting.

This lack of efficacy is most noticeable to players when actions they take in the world and those which they are told they have accomplished are not reflected back to them. After blowing up 30 dyspeptic alpacas (to collect 3 hides) the world looks the same. Dyspeptic alpacas (or ferret kings, or fel hobos, or goblin chiropractors) continue to spawn at pre-determined rates, and NPCs neither remember what you might have done for them (outside a quest chain), nor have heard of you (regardless of how far you have progressed in the world narrative or what you have supposedly achieved). And (as pertinent to quests as a narrative device) this disconnect between player action and narrated action actually discourages players from reading quest content or really engaging with the world narrative (since it doesn't engage with them back). When all content is reducible to a repetition of the ‘kill something and return with a macguffin’, the exact reasons for killing a specific enemy become vanishingly irrelevant; the content is only a vehicle for objectives and a reward. Even the graphic design of the quest interface, particularly in World of Warcraft (see below, in Figure 1), directs attention away from the narrative. By using contrast to pull the reader’s eye downward, toward the objectives and rewards, the description is skipped over entirely, short of whatever contextual words happen to appear in the highlighted portion of the texture.

---

Addressing these flaws in player experience would seem to require two paths of resolution, curative and palliative. Efficacy and meaningfulness of choice here are almost trivial problems to address, from a palliative standpoint. To draw an example from *World of Warcraft*, heavy use of multi-stage instances in *Wrath of the Litch King* (See: Wrathgate, The Battle For The Undercity, etc) shows how player action can be used to visibly affect the world in a way that is both interactive and cinematic (in the expansive, visually dynamic sense) to each player, as they reach appropriate narrative checkpoints. The world changes enough for players to feel as though their actions had meaning, and the narrative becomes richer for that interactivity and permanence.\(^{31}\) NPC notoriety is an even more trivial way to make players

\(^{31}\) As seen, for example, in NCsoft’s Guild Wars series. Guild Wars is entirely instanced-based; and each instance has its own narrative arc, with a terrain that visibly changes as you complete various tasks.
feel respected and valued in the world; the flagging and quest-modifying text variables (to detect reputation tier changes and modify barks according to race and class) already exist, and would merely need to be applied to narrative checkpoints reached by the player (data already being polled for achievements). And by also letting players have a stake in the creation of in-world quests (giving them some involvement in the flavor of the world if not the 'official' content), the players themselves can provide a palliative treatment for some of the boredom associated with re-play. Here we can look at how the Mission Architect allows players to provide XP and Reputation rewards to one another without infringing on the narrative of City of Heroes.

To 'cure' the problem, however, we must encourage content design in which the player engages in interesting or complex actions instead of repetitive ones. We need to find alternate means of gating access to rewards and achievements, rather than encouraging players to engage in tedious or game-world-breaking behavior. Here we can look to virtual worlds (like Second Life, There.Com, Uru, etc) for examples of how to design player tools for placing objects and interacting with those objects; but more importantly, for examples of how players might use those tools to create their own multiplayer narratives.

As Storytelling Device

While there are clearly significant mechanical and structural goals associated with quest design, we nonetheless think of quests in MMOGs as the primary distributor of narrative and the dominant method of storytelling in those games. Within that context, what is the (diagetic) purpose of a quest?

- Involve the player in the world narrative
- Emphasize individual action in the scope of a multiplayer universe
- Tell an atomic story with a defined arc
- Which fits into a sequence of similar arcs (of greater or lesser complexity)
• Which, taken sequentially, tell the story of an zone (which, semi-sequentially, composes the story of the world)

As we’ve seen in the previous two sections, quests must serve a variety of other requirements before they can begin to tell a story. We’ve seen that in practice, this is often through a task structure with adequate content to contextualize task ("Can we solve this problem with violence?" "Yes we can!"); followed up by exposition rewards which are designed to be glossed over. Setting aside those expectations, however, how might we use the structure of quests tell better stories? Broadly, we might consider a story with an episodic structure (addressing the scoping goals of a unit of interaction). This story would be as interactive as the medium allows (addressing the game play goals of a unit of interaction). It would engage more than trivially with the environment in which it takes place (addressing the locative goals of an environmental unit of interaction). It would be, at least in part, about the player's character (addressing the efficacy and contextual goals of an environmental unit of interaction). And it would reveal an aspect of the world or a character (addressing the contextual and sequential narrative goals of a unit of interaction). Those final two goals have been approached in two major ways in MMOGs: player choice (with shallowly branching narrative), and cinematics (attached to significant quests performed in-world). Live events can be considered a special class of cinematic, and will be discussed briefly in the "Cinematics and Film Envy" section.

The Myth of Player Choice

The underlying theory behind introducing player choices in a single-player narrative is that it will help players engage better with their place in the world’s narrative. However, because MMOG content cannot meaningfully branch, player ‘choice’ is primarily a question of statistics. A player chooses reputation with one group over another or one item over another, according to how that choice will

32 "Interactivity is almost the opposite of narrative; narrative flows under the direction of the author, while interactivity depends on the player for motive power" (Ernest Adams, "Three Problems For Interactive Storytellers". Gamasutra)
affect their combat statistics. Which tends to result in quest design where a player’s primary decision is whether to take the item with a +8% attack power or the one with +6% attack penetration; without real regard to the decision being made (as it won't manifest in the world).

Figure 2: Three views of Shattrath, the new major city associated with the ‘World of Warcraft: The Burning Crusade’ expansion. The two factions introduced have separate ‘sections’ of the city.

For example, let’s compare group membership functions in two separate games: the ‘Aldor’ or ‘Scryer’ choice in ‘World of Warcraft: The Burning Crusade’, and the ‘Connected: X’ group membership options in Echo Bazaar33. Echo Bazaar is a Twitter-based ‘social’ game, which uses an action points accumulated over time model. Echo Bazaar gives players choices with narrative payoffs (if I acquire story paths A and Connected: C, path E will unlock; etc) that are rarely exclusive, but which are their own reward (players perform tasks for more narrative rather than more stuff). Wow:tBC gives a choice that seems simultaneously superficial and significant: both groups have similar goals, and inhabit the same city, but the group you do not choose will become hostile when you make an otherwise context-free decision. The decision made in Wow:tBC is purely statistical (the sets of bonuses associated with each faction is clearly superior for a subset of classes), providing some palette swapped clothing, and, most damningly—no long term consequences. A Horde Undead can join the Aldor (represented by the Alliance Draenei); and an Alliance Gnome can join the Scryers (represented by the Horde Blood Elves). Neither will receive censure from their own ‘home’ faction, or different behavior from other, non-faction members of that race. Non-Aldor Draenei NPCs are no less friendly to Scryer aligned Alliance, and no more friendly to

33 A twitter game, produced by Failbetter Games.
Aldor aligned Horde. Let's contrast this with the narrative structure of social game *Echo Bazaar*. The decision made in *Echo Bazaar* is purely narrative (opposing groups, such as The Constables and Criminals only force decisions when reputation with both groups is very high).
While these games seem very different, they have fundamentally similar flaws. In *Echo Bazaar*, the reward is narrative and in *WoW:tBC*, the reward is stuff, in neither case is the choice actually significant. Players cannot experience true consequences for two major reasons. First, player expectation: No current MMOGs prevent players from ‘starting over’, and as a result, players frequently ‘change their minds’ in order to experience a greater variety of content (or get more stuff), so access to that content cannot be truly cut-off. Second, content is finite: Designer-created content can only be produced at a fixed rate, which will always be outstripped by the rate of consumption. Players need to be able to go back and change sides to be continuously experiencing new narrative content. Introducing significant consequences or narrative branching requires an exponential quantity of content development, and works against existing player expectations; neither of which is beneficial to player retention (and consequently, sales).
A fine day in the Flit
The Flit teems with opportunities for the crafty and the quick.

Go hunting for caches
People hide things in the Flit, often on ropes under bridges and gantries. The Raggedy Men call the business of stealing them 'fly-fishing'.
(A straightforward challenge for your Shadowy quality. [7])

Spy on the Raggedy Men
The beggars of the Topsy King's court are a strange lot. But they rival cats for their knowledge of secrets.
(An almost impossible challenge for your Shadowy quality.
You have been warned., [7])

Spy on the Raggedy Men
The beggars of the Topsy King's court are a strange lot. But they rival cats for their knowledge of secrets.
This choice is locked for now.
To unlock, you need Ambition: Heart's Desire – the Topsy King 3.

Pretty flames!
Most of the dockers live in the stalagmite fields behind Wolfstack Docks. You toss a match through the window of a rickety hovel, and the place goes up like a firework. Voices cry out in horror and alarm bells split the night. You stroll away, sooty-faced and grinning. A troop of Constables sprint past you without a glance.

You succeeded in a Dangerous challenge! [7]

Dangerous risk increasing... [7]

66 67

You now have 69 x Rostygold [7]

* A Neddy Man has increased to 10! [7]
Our techniques for avoiding shallow player choice seem limited within the context of the quest model. *Echo Bazaar* might be pointing in the correct direction, however. Contacts are also a form of currency in-game, and can be turned into goods needed to unlock other storylets. As the (very small) body of designers rolls out new content, players have significant motivation to replay old content in order to unlock all of the content in the current ‘Book’ of storylines. By making the narrative itself a reward, and minimize the time ‘grinding’ takes to complete, we can at least lessen its game-breaking qualities. The lack of consequence, however, is a much trickier problem to solve. Consequence is necessary for deep storytelling, but is deep storytelling necessary for fun or game play? What role should this storytelling play in the narrative design of an MMO?

**Cinematics and Film Envy**

As discussed above (in “The Myth of Player Choice”), there is frequently an absence of actual 'role-playing' in the mechanics of MMORPGs. Instead we find a proliferation of 'RPG Elements'- binary (or ternary) choices, limited character customization, etc. This allows players to make small decisions about their character, without requiring the explosion of content discussed above. And since (as discussed in “The MMO Quest: As Unit of Interaction” on page 19) we also rely on 'quests' to tell non-meaningful 'learn to play the game' stories, players are actively discouraged from paying *any attention* to quest content. In other words, to preserve game play we find ourselves first removing any context or consequence for character behavior, and then actually divorcing player and character behavior. Content writers are necessarily finding themselves with less and less to work with, and MMOGs are coming to rely on what was once a sing-player game convention; using cinematics to tell stories about other people34, or to tell the player who their character is35 for them. It’s easy to see the temptation. Where

---

34 For example; in World of Warcraft: The Wrath of the Litch King, The Battle of Angrathar the Wrathgate, which is primarily about Dranosh Saurfang, Bolvar Fordragon, and Grand Apothecary Putress.

MAST (Metaphoric Adventure-Scripting Tool): A user facing tool for creating multiplayer MMO narratives

Audrey Whitman
player behavior cannot be assumed to be diagetic in any way, and the choices that players make must be reversible; in order to provide any sense of a coherent world, or control the flow of narrative about that world, it seems necessary to remove players entirely and focus on characters that were created for that narrative. And, in the rush to 'cinematize' games and provide them with ever-increasing degrees of visual fidelity (supposedly giving them more cultural legitimacy), writers find themselves taking control away from the player more and more, in order to insert story around the assumed primary mechanic (of applying bullets to the world until plot comes out). Unfortunately, because it takes control away from the player, it encourages a 'receptive' rather than 'participatory' relationship with the narrative, disconnecting the player from their character. And in MMOGs, where sovereignty of player character control is a fundamental game play feature\textsuperscript{36}, 'cinematic' play is not play at all.

Ultimately the biggest flaw here is conceptual; it relies on the visual story-telling mechanics of movies, television, or comic books to fill in the gaps of storytelling produced by the deficiencies of the quest model. The better solution would be to emphasize the mechanics of interactive storytelling; but that is trivially accurate. If interactive storytelling were the 'silver bullet' of MMOG narrative, everyone would be doing it. So let's unravel a bit, and look at some mechanical and structural reasons why they might be so common:

- To wallpaper over actions the system cannot perform (i.e. render at a specific depth of visual fidelity, or stagger spawns and barks in an engaging way for the player).
- Less frequently to show some bit of story as a coherent scene between two or more NPCs, (which may or may not involve the player).
- Or finally, a 'reward' for a level or a narrative checkpoint reached (rarely used in MMORPGs).

\textsuperscript{35} For example; in Aion or Guild Wars, which begin in medias res, eventually telling the story of your character’s role in the world at large.  
\textsuperscript{36}If your primary mechanic is real-time combat, every time the player loses control over their character, unless a cinematic literally removes them from the game world, they are at risk of unexpected combat encounters.
So, to avoid using visual storytelling (as we say above) while still telling engaging stories which use the space they exist in requires a surprising degree of vigilance. Designers must be mindful about not writing themselves into narrative corners; always relying on actions players can perform in a space to tell those stories. Second, designers must deliberately create the circumstances that would produce the behavior desired for a given piece of interactive storytelling. Rather than trust the player to behave in a certain way, or prevent them from acting otherwise by taking control over their character, the context of the narrative itself must provide the motivation. As an example I'll work off a fairly famous cinematic from *World of Warcraft: The Wrath of the Litch King*, The Battle of Angrathar the Wrathgate; re-tooling it as an interaction-based story. In Chapter 4, I will walk through an extended example of how this sort of interaction design might work in *5 Boroughs*, using MAST.

![Figure 5: Two 'scene' segments from the 'Battle of Angrathar the Wrathgate' cinematic.](image)

The Battle of Angrathar the Wrathgate has all the earmarks of a 'necessary' cinematic: the story is told through the voices of four 'significant' character NPCs, and involves a pitched battle with multiple clear, visually distinct stages. A player character wading into the midst of that battle (were it to be positioned in the live game world) would find it difficult to 'see' each of the cinematic scenes, and would lose the camera focus and staged clarity of each of those sequences. However, this is actually far less damning that it looks: each of these points is an opportunity for adding interesting player interaction: "Whatever their variety or the name we give them (quests, missions, adventures, exchanges, errands, tasks), quests
are the chance for the game designer to bring the storytelling elements into play. And if there should be any general recommendation for designers, it would be that they try to entwine structure and story as much as they can in their quests.¹³⁷

Figure 6: Two scene segments from the 'Battle of Angrathar the Wrathgate' cinematic, featuring world narrative characters Dranosh Saurfang (Saurfang the Younger), Bolvar Fordragon (the Highlord), and Arthas the Litch King.

Simply letting the players join the battle with their own side (as is later done with The Battle for the Undercity), the fight feels more urgent; suddenly the player can affect the outcome of the conflict (in however small a way). Fairly trivial location sensing could then give players a front row seat to each of those sequences- how much more tragic does the death of Dranosh Saurfang feel when the player is right next to him when it happens? How much more threatening is the Litch King's emergence when the player is fighting their way toward the opening gate when it occurs? But even better, let these events happen to the players as well as the NPCs. Rather than simply watching Bolvar Fordragon succumb to the plague, create a drop-in group (as seen in battlegrounds) where I can see other players succumb to plague, and give me special actions to attempt to revive them or pull them to safety (according to class, race, etc). Make it my responsibility to help win an encounter, and I have a much larger stake in the outcome of events. And borrow interactivity from the cinematics of other game genres: let my character

interject dialogue (if I chose) into a scene, let me use my in-game abilities (such as they might be) to take actions within a space where story is happening.

Figure 7: Two scene segments from the 'Battle of Angrathar the Wrathgate' cinematic, featuring world narrative character Grand Apothecary Putress and the Red Dragonflight

To see these techniques in another context, imagine a staged attack on a player-friendly town. Players can drop into a 'live' (or, more likely, instanced) event, with players in a similar level bracket, to defend the town. Entering certain 'zones' of the town (airspace, tower, gate, etc) opens certain defensive options that can be performed while in those zones. While some players engage the attackers, other players can build up or restore broken defenses in these zones, providing buffs or other bonuses to the fighting group. Named NPCs can enter the conflict at various points, spawning local events that let players defend or rescue NPCs, or have brief conversations with them.

Cinematics, in comparison to consequential storytelling, are much easier to address mechanically. Given some premise of what interactive narrative in an MMO might look like, we can summarize our techniques as: render it in the live game world, let it happen to the player instead of an NPC (and if it can't happen to a player, let it happen when the player is located near a narrative 'event'), and always incorporate actions the players can take to affect the outcome of an event (even if the effect is minimal). By drawing many forms of interaction into the game space, we can bypass most of the
traditional rationales (or requirements) for including cinematics, by creating opportunities for players to act and engage directly with the narrative of a space.

**User-Created Content**

As we’ve seen, narrative engagement through questing is hard to put our finger on, so to speak. The model itself undermines a designer’s ability to write stories that make sense in a multi-player context. Player (or user) created content is frequently held up as the solution to this problem. It eliminates some of the content pipeline issues associated with top-down content control, by providing a theoretically inexhaustible supply of player to player content exchange. And within the context of a given player-produced 'story', it is possible for a player's character to make meaningful decisions. There are several problems that emerge in approaching this as a viable strategy, however. While it may be democratic to open a game completely to player control, it quickly becomes impossible to maintain any sort of canon control. But if player content is wholly divorced from in-game significance, player interest in performing that content will be dramatically reduced.

So how do we get players to care enough about the stories told in-game to want to tell their own in that same space? How do we encourage players to engage directly with the story embedded in the game world? The tool may be fun, but no one will find that out if there are no in-game incentives associated with using the tool in the first place. We remember that our keys here are emphasizing a player's actions in a given space (and remembering those actions over time), structuring the narrative such that it is rewarding for players to perform (as opposed to merely divided into hooks and rewards for completing a related task), and by rewarding players for 'expanding' the world with their own stories. Turning those requirements for a designer produced single-player story toward my own work, let's look at the role player-created content might have in the context of *5 Boroughs*. 
In our initial design concept, we wanted to provide a balance between total canon control (World of Warcraft) and no canon (Furcadia, Second Life). So, drawing from the model of City of Heroes & City of Villains, in 5 Boroughs, content would be private (non-canonical), but sharable. The primary distinction we draw from CoH's model is the relationship between content creator and content user. In City of Heroes, the only avenue of contact between consumer and creator is through the 'Star System', a rating and commenting tool that delivers optional, anonymous user feedback in the form of a 1 to 5 rating plus comments. So while content creators may receive feedback, the design of the feedback system produces vague advice, divorced from contact with the player who left it. In 5 Boroughs, however, members of a group can share created content with other members of that group, which they can perform for in-game rewards (both player made items, and procedurally generated reputation). This sharing scales from the smallest player group, your 'family', up to larger organizations and associations based on the feedback of the other members of those groups. Produce stories your family (composed of your in-game friends) likes, and you can produce stories for larger groups of people. Story production at the faction level (the largest group size) can be submitted for designer feedback, to be placed in the 'canon' world, allowing members of other groups to see and engage with the best narratives produced by their fellow players, in addition to those produced by their own social circles.\(^{38}\)

Player-produced narratives, or stories, are meant to bridge the gap between analog role-playing (with your friends, around a table or through a computer screen, in a world of your own devising) and digital role-playing in worlds which are not responsive to that play. They will allow players to continue telling stories to and with their social circle, through a mechanically driven game, within an extensively designed virtual space. By letting players tell their own stories as well as the stories we have produced for them, we hope to encourage players to become active in content development at all levels of the game. Because, these same groups will also be able to band together to perform multi-player canonical\(^{38}\)

\(^{38}\) http://egg.lcc.gatech.edu/wiki/doku.php?id=ellisisland:spring10demo_mmonotes_031210
narratives (world events) created by designers, and through affecting those events, affect the world for all players. So by scaffolding player-generated content (starting with personal narratives, eventually scaling up to large-scale narratives), and providing a context for that narrative which players can also affect; we believe that players will then want to deliberately produce content which can ascend to the level of a world event. In a finished version of 5 Boroughs, this tool would become accessible during the 'leveling/tutorial' process, and the player would receive tasks that require and teach its use. By the end of this process, players will be proficient in their use of MAST, and prepared to use it in the multi-player endgame.

**Comparison Studies**

Now, how can we draw those techniques into a player-facing tool? First, let's compare against existing tools which attempt to answer the question of letting players insert interactivity into the world. The following section will quickly review several similar tools, and touch on the problems raised by them.

**Mission Architect**

The Mission Architect is a tool for developing custom 'missions' or quests in the *City of Heroes & City of Villains* universe. Diagetically, these are represented as 'training exercises', which players can be rewarded for with experience and 'tickets' (item tokens).

![Figure 8: Architect Entertainment Hologram Contacts & Data Stream](image)

*Figure 8: Architect Entertainment Hologram Contacts & Data Stream, the 'interaction point' for starting a player-created mission. The contact's appearance can be modified in the mission creation process.*
Players use a point-and-click interface to create story arcs of up to five separate missions. Customization involves: selecting options (size and type of the map, mission objectives, encounter types, difficulty level of each encounter, choosing the powers and costumes of some of the foes, etc) and filling in all the text of a mission (of which there is comparatively little).

![Figure 9: Example screenshots from the Mission Architect interface (there are 14 screens in total).](image)

**Similarities & Differences**

While the Mission Architect should be the closest model to MAST (also developed for an MMOG, involving a point-and-click interface), it is functionally speaking a very high-level level editor. Players design a miniature instance, fill it with things to kill, and provide customized barks and skins for those things. It would be fairly trivial to re-skin a tool like this for another 'kill stuff placed in the environment' game (e.g. *WoW*, *Everquest*, etc). The tool is, however, essentially representative of designer-produced material in-game (in terms of complexity and player interaction).

**Pertinent Design Influences**

Initial versions of my design strongly resembled Mission Architect screens. Ultimately, I found them too cluttered and visually distracting, and far too focused on building a story out of a set of existing parts. I wanted to focus MAST on a more 'free-form' gm-moderated style of play (as opposed to game-moderated play).
Spore: Galactic Adventures

*Spore: Galactic Adventures* added the Adventure Creator, a tool very similar to the Mission Architect. It allows players to create missions, which are then populated into the Space stage of Spore. The interface is drag-and-drop; allowing players to add any objects (creatures, vehicles and buildings) previously created or downloaded to a level. There is some minimal creature behavior scripting (aggressive to friendly), ‘special effects’, sound, and dialogue (through speech bubbles). A complexity meter exists to prevent too many objects being dropped into the game.

![Figure 10: Several screenshots, including Spore Galactic Adventure levels and some of the level editing tools.](image)

Similarities & Differences

Fairly similar to both the Mission Architect (discussed above) and the Electron Toolset (discussion to follow); *Spore: Galactic Adventures* stands out for its simple interface and editing capabilities. While ultimately still a poor fit for the goals of MAST (it is still primarily a tool for designing spaces rather than content), it distantly engages with the idea of multiplayer activity.

Pertinent Influences

In reviewing adventures created using the Adventure Creator, I was surprised by how many level modules left the text options in their default state. When given the opportunity to write dialogue (or even staggered barks), many players will opt not to. It occurred to me here that in presumed multiplayer content (like that produced by MAST), dialogue was not only largely unnecessary (since NPCs could be
run 'live' instead), but actively inhibiting (by adding more weight and complexity to the tool). I removed dialogue (outside of trigger-able 'Events') from further iterations of the design as a result.

**Electron Toolset (Neverwinter Nights 2)**

The Electron Toolset (previously the Aurora toolset) is a tool for developing *Neverwinter Nights 2* mods and custom levels. Unlike the Mission Architect, the Electron Toolset is a 'true' level editor, with terrain and mesh editing, designed specifically for D&D 3rd edition-backed rpg play. Technically modules can be created for persistent worlds, but the toolset does not support this well, and single-player or multi-player is preferred.

![Figure 11: Sample screenshots from Neverwinter Nights 2 and the Electron Toolset Terrain Editor.](image)

Significantly more complex narratives can be created using the Electron Toolkit, but they are almost exclusively single-player, and tend to fall heavily within the Monomythic quest model discussed in the "The MMO Quest" section, on page 14.
Similarities & Differences

Obviously the Electron Toolset is too heavyweight an editor to compare directly to MAST, and the central premise (role-playing, mediated by the game rather than by a GM) is explicitly the opposite of MAST's design goals. However, while it is not a online game editing tool, tools like the Mission Architect and Spore Galaxies refer to and build on the conventions set by this series of tools. Moreover, NWN2 has a very active online modding community, which is very necessary for the continued life of a content-creation tool. New player involvement in a community of practioners requires an active forum composed of players at various experience levels, which keeps the pool of active 'creators' refreshed. 5 Boroughs, by extension, needs to cultivate and maintain its forum community to keep MAST in active use among its players.

Pertinent Influences

In the process of designing MAST, I learned to make a fairly important distinction between GMs (people who do casual world-building and group-oriented storytelling, using a mechanical system of their choice), and prospective level designers. The more complex and structured the tool, the more clearly it falls in 'level design' territory, and the more difficult it is for a novice user. MAST needs to be easy

---

enough to grasp that a GM can get up and running sessions in less time than it would take to prepare for a session of a tabletop game; with no more than a few hours (as opposed to days or weeks) spent learning.

**Conclusions**

These three ways of looking at a quest (as a unit of interaction, as an environmental supplement, and as a discrete unit of narrative), let us address the practical concerns associated with quest design. They allows us to design quests which (in their platonic state) serve both the mechanical and narrative needs of a single player, while invisibly fleshing out the narrative of an NPC, a zone, or the world. However, even these platonic quests, re-analyzed through the lens of a multiplayer world, fail to meet the needs of a truly multiplayer game.

Given the previous discussions, when we acknowledge the failings in quest design, we think of them as failures of scope, direction, or player action: lack of focus on player experience, lack of player 'efficacy' (with respect to either the environment or NPCs), or an overemphasis on statistical rather than narrative choices. We think within the quest model (made famous by Everquest) which is predicated on repetition; and which game designers have struggled to 'patch' with the storytelling tools of traditional media models (e.g. film in cinematics, print in comic sequences, etc) which are ill-suited to addressing the real narrative gap; completely missing the intrinsic reliance on the player acting *alone*, solo participant in their own quest. While it is possible to simply refine the design of quests to better serve both their mechanical and monomythic underpinnings; those quests would still suffer from the same flaw discussed above. They would remain single-player content for a multiplayer world. "This is the key
to creating interactive storyworlds: multiple but connected themes. An interactive storyworld must present the possibility of romance, betrayal, battle, spiritual growth, and many other possibilities."\(^{41}\)

Chapter 3: Artifact Design

This chapter follows the evolution of MAST; a user content creation tool which, informed by my academic research and industry experience, aims to address the problems discussed in the previous chapters. It documents my initial design assumptions, how they were revised over successive iterations of design, and annotated with the various insights I had along the way. As has become obvious over the course of the design cycle of this artifact, this master's project became far too large to develop by a single person in the available timeframe. In trying to fully grasp the problem I was attempting to address, I needed to iterate through progressively more challenging programmatic tasks, eventually creating one which I could not successfully complete (except in a non-functional, screen progression form). As a result, I moved toward a more purely representational task; to fully document the functions of this artifact (with a full walkthrough to follow in Chapter 4), and show how information would be shared through these tools. This document, then, is better considered a formal design document (explaining the format, programmatic underpinnings, and goals of the tools); which is meant to stand on its own as something from which these tools could be built.

Introduction to MAST

When a GM plans a game for a group of players, they plot broadly; taking a concept, comparing it against a narrative direction established by player behavior, and breaking it down into a sequence of short-term events. These events may be sequential, asequential, or both; but they emerge in actual play episodically, in response to where players direct a story. Essentially, the GM and the player aren't actually playing the same game. The GM is orchestrating the delivery of narrative events, the placement and frequency of combat encounters, and refereeing interactions between players. The players are attempting to find ways to use their skills in the context of the game world, while meeting their own
narrative objectives. To allow traditional role-playing to transition into digital space, our tools need to first recognize that the GM continues to have an active role during a session of play (rather than building a clockwork for the players to interact with on their own); and that this role is not identical to that of any other player. A useful set of tools to meet the needs of a GM might include a pre-game 'planner' which prompting the GM in useful ways, and an in-game RTS, letting GMs direct players, give them clues, and moderate the difficulty of events.

How can we use the model of 'unit of interaction' to think about how players interact with narrative and world? MAST, as has been discussed before, is a design prototype of a user content creation tool; specifically, a narrative content tool, or a 'quest builder'. While MAST is being built in the context of 5 Boroughs; it was designed according to a process I established (as a part of my design cycle) necessary for the development of narrative content tools. In this chapter I will discuss that process, how I used it to develop the preliminary design of MAST (the feature list), and how it continued to influence the development of MAST over the iterations of design discussed later in this chapter.

**Design Process**

The design of MAST was a process I undertook in stages: beginning with the broadly defined goals of the system, from which I established my 'design principles' and a clearer understanding of the system's needs, enabling me to create a useful example user story, from which I could extract a preliminary feature list. With those features defined, I could begin to design the eventual tool around those requirements, and produce the screen breakdown shown later in this chapter.

The central question of a narrative content tool for an MMOG is where user-created content exists with respect to the diageic world of that MMOG. In 'themepark'-style games like *World of Warcraft*, that player content is (regardless of its internal diageic EXTRA-diageetic state) entirely external from the 'canonical' world. Narrative created through the available tools (public and private chat, forums, in-
game materials, etc) is not 'stored' by the game, or persistent in its diachronic world. Partially themepark-oriented games like City of Heroes/Villains allow player content to exist and be 'stored' by the game, but as non-canonical 'training missions' which give real rewards, but did not actually 'happen'. On the other end of the spectrum, MMOVWs like Furcadia, Second Life, and There.com allow a significant spectrum of content production on the part of players, but have no world canon or context for that content to exist within; both content and context are wholly provided by the player.

I believe that players will want to engage with MAST in this way for two reasons: as described previously, MAST is an unusually responsive narrative tool for an MMOG/VW (as compared to City of Heroes & City of Villains above), and (in intention, at least) conceptually simpler than other visual story builders (represented here by Spore: Galactic Adventures and the Electron Toolset). The combination of narrative freedom, and careful scaffolding will provide an optimal test case for the usefulness of user-produced content as a method for improving and streamlining the content pipeline of a game development company. The core assumptions that we must make about our player base are that they are engaged in the narrative of the world, invested in the process of telling their own stories, and willing to expend non-game-play time to develop those stories using a tool built from the underlying mechanics of the world. The players who will seek out this tool will not, as a whole, be engaged in the minutia of trade politics for example, or in planning out the literal guard distribution of a one-block office complex. Their focus will be on stories, which occur with the player (or players) at their center, and on interactions that players can have with NPCs and one another over the course of completing a story. Set-pieces and dialogue, in other words; a movie set for players to travel through, stopping occasionally to see the sights. In the case of 5 Boroughs, maintaining the metaphor of the game itself (storytelling through life experience, mediated by tarot cards), creates a visual 'dictionary' of actions that are unique and evocative, providing a valuable layer of interpretation to that player created content; and we believe that additional creative flexibility will be a significant draw for players.
Design Philosophy

Since we are encouraging players at all levels of experience to be producing stories for their in-game friends, and relying on emergent behavior to drive world PvP, it's urgent that these development tools meet the following requirements:

- Be accessible to players with low programming experience
- Be visually clear (simple, un-cluttered UI)
- Promote multistage story telling (through information dense 'template' blocks)
- Integrate well into the central metaphor of game-play

These template blocks must:

- Provide natural room for player customization
- Provide thematic structure for creators
- Guide creators in setting story goals, and breaking stories into discrete pieces.
- Assume (and build in the background) drop-in multiplayer— In other words, multiple people performing the same (or related) tasks created by the same story teller, merge into one instance, and can assist one another without needing to deliberately join a group of players to play through a piece of story together.

Structurally, narrative performed in (an instanced version of) a game world is complex to develop, and only more so for writers (and storytellers) without significant scripting experience. To ease that learning process, the development tools need to have extensive scaffolding for beginner content creators to 'learn' the language of the system; but they also must have the freedom to 'take off the training wheels' and develop on their own. To reconcile this design problem, I started with what I knew about the 5 Boroughs project; and, from that thought process, extracted the following design principles.
There are no 'rules', per say, in this suite of design advice. It's more like a process; a design gedankenexperiment which forces the most pertinent qualities of the artifact you are designing a content system for to the forefront of your mind. There isn't a set of absolute guidelines, pertinent to all forms of content development, across all types of online games. Content production for an MMOG/VW with a complex 3D environment is necessarily structurally distinct from content production for a purely text-driven MMOG/VW. But irrespective of differences in theme or medium, there are central questions that must be addressed:

1. What is the IP (or if not the IP, the story world)?
2. What is the core of its emotional experience?
3. Why will players want to produce content for an MMOG/VW based out of this IP?
4. How will you incorporate feedback & content (from your development team to players, and from players to your development team) into the world?
5. What form will that content take, in-world?
6. How will you encapsulate content (what is its scope)?
7. What will content consist of (what are its pieces)?
8. What is the central mechanic of that IP/idea?

Abstractly, content produced for a multiplayer world must: relate to the core emotional experience of the world, through the mechanisms of that world, affording the types of interactions players expect, which the world responds to in predictable ways, and which is sufficiently modular as to allow players creative freedom in developing that content. "The role-playing game exhibits a narrative, but this narrative does not exist until the actual performance. It exists during every role-playing game episode, either as a memory or as an actual written transcription by the players or game master. It includes all the events that take place in character, nonplayed character backstories, and the preplayed world
history. It never exists as a code independent of any and all transmitters, like Schechner’s definition for
drama suggests.”

To begin laying the groundwork for a design, I wrote myself a sample user story, and attempted to
derive a list of necessary features from it. What follows is that user story, and the impressions I derived
from it.

Example User Story

The player (Eliza), wants to create a story hook for a role-playing group she’s managing through 5
Boroughs, about Little Timmy Tenspot, a newsie from the Bronx zone. Timmy Tenspot is an NPC created
by Eliza as an information source for her role-playing group as they research the mysterious death of a
newspaper heiress. But to secure his assistance, they must first build his confidence through other tasks:
help him conceal his little sister from an orphanage, discover an unknown relative in the city, deliver a
letter from him to that relative, etc.

In order for our sample player Eliza to use MAST to tell an ongoing story in 5 Boroughs, what does MAST
need to be able to do?

1. Player/Character Flagging States:
   a. Staged-narratives allow for more narrative complexity.
   b. By tracking and saving character states over several pieces of a larger arc, GMs can track
      player progress more accurately, and direct players more efficiently.

2. Minimal Interface:
   a. A lightweight interface will help reduce learning difficulty.
   b. Reducing the learning curve is necessary for encouraging players to use MAST.

3. Templates

---

a. To encourage fast adoption of the tool, it should include templates provided by designers (as well as let players create their own templates for re-using stories),
b. This will provide examples of how to produce effects, and may make it easier for new players to create content in the first place.

4. Provide Tarot-Card Focused Storytelling
   a. This will link storytelling in MAST to the mechanics of 5 Boroughs, reinforcing the interpretative metaphor.
   b. The narrative density of tarot spreads allows us to incorporate spatial & symbolic meaning; and information density allows complex narratives with many pieces to remain easy to navigate and examine pertinent details of.
   c. By allowing placement & symbolism carry weight we don't eliminate the learning curve of a development tool, but we reduce it.

**Design Process**

I approached the design of MAST with two central goals:

- To create a tool for user-created content which reinforced the impressionistic, emotionally realistic world narrative of 5 Boroughs, and relied on its central play metaphor of Tarot cards.
- To create a tool which emphasized the social relationship between GM and player in face-to-face tabletop role-playing games, and allowed GMs and Players to interact meaningfully with one another through play.

The tool, furthermore, should:

- Require minimal scripting
- Allow for freedom in expressing 'non-traditional' MMORPG narratives
• Have a flexible information architecture which allowed for 'on the fly' modifications to storytelling data

• Clearly demonstrate the benefits associated with online play (knowing that tabletop GMs would not necessarily be immediately friendly to translating their games into a digital space)

Working first from my own experience with content creation tools, I attempted to identify the potential conceptual hurdles for a new content developer, and to hypothesize interface design solutions.

I began to break down a story into component pieces: Arcs, Scenes, and Events. A scene required one event, which would be a resolution state. A resolution state would be a set of conditionals, structured like madlibs, with a selection of conditions (e.g. 'any player does'), qualifiers (e.g. 'if', 'when', 'and', 'or', etc), and actions (e.g. 'advance arc', 'reputation increase', etc). Resolution states could be triggered by events.

Unfortunately, while it represented a step in the right direction, this design still relied heavily on scripting, prior knowledge of Tarot, and forced GMs to invest a significant amount of time doing preparatory work which would remove their direct role in the story. For genuinely reactive (though not participatory) play, that content would need to be constantly revised, making the tool itself a time sink for GMs, and something of an impediment to group play. While it might make an interesting narrative-focused development tool (e.g. Wintermute, Neverwinter Nights, etc), it wasn't a roleplaying tool.

The design of MAST proceeded iteratively, with several 'initial variations', which eventually lead to the metaphor which defined the last leg of design development: Stories, which consist of arcs, which are made up of scenes, which are made up of events. Alongside, we have sessions, which contain a single 'instance' of group play. By relying on this narrative metaphor and the visual metaphor provided by tarot cards, I hoped to create a creatively flexible and game-like development environment.
**Architectural Design**

The following page shows a map of interaction between the five 'moving pieces' associated with MAST, demonstrating how information passes from artifact to artifact. To describe briefly, in the center of the illustration, you see two overlapping circles, the game itself (5 Boroughs) and the database underpinnings. Around it are three boxes representing MAST (the out of game GM planning tool), and the GM & Player Tools (the in game tools for GMs and players). MAST primarily receives data from the database (in the form of menu contents and generated information); and sends data to the GM Tools (in the form of actions the GM can take in-game. The GM Tools primarily receive data from MAST (as above) and 5 Boroughs (in the form of player states); and send data to 5 Boroughs (in the form of NPC updates, event triggering, and locational changes) and the Player Tools (in the form of changing NPC states, dynamic location information, and static information (released by the GM) delivered to the player. The Player Tools primarily receive data from The GM Tools and MAST (as above); and send data to 5 Boroughs and the database. 5 Boroughs primarily exchanges data with the database; and receives status updates from the GM Tools and Player Tools (as above). The Database primarily exchanges data with all other pieces (sending & receiving updates to stored information); and reconciles changes and returns updated information.

To clarify a few terms used in the following image:

- An example of 'Player Data' seen in the following image: a character's name and statistics, their x/y/z position in the world, whether they are a GM, etc.
- An example of 'Static Data' would be player notes, or descriptions of locations.

---

In the name of strict accuracy, what I am referring to as 'the database' here encompasses both the literal database and the scripting shell around it that processes and queues information requests and updates.
• An example of 'Event Data' would be trigger conditions, NPC state changes, location information, etc.
Figure 14: Data flow chart of MAST, 5 Boroughs, the GM Tools, and the Player Tools.
Final Design Iteration for MAST

After several false starts, I returned to the conceptualization phase: What was a story? What kinds of stories might GMs want to make? What kinds of stories might players want to play?

Returning to the question raised early on in my design process; I needed to find a way to emphasize the social exchange between players and GMs during a session of play, and create the circumstances that would allow GMs and Players to interact meaningfully with one another through online play.

![Minimal (Story Name) | Sisters of Mercy (Arc Name)](image)

**Arc Information**

**Sisters of Mercy**

<table>
<thead>
<tr>
<th>Dramatis Personae</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Nancy (Mafioso)</td>
<td>This arc is about terrible references to band names, science, and blood-sports.</td>
</tr>
<tr>
<td>Eddy (Politician)</td>
<td></td>
</tr>
<tr>
<td>Missy (Artist)</td>
<td></td>
</tr>
<tr>
<td>New Player (Player Class)</td>
<td></td>
</tr>
</tbody>
</table>

**Previous Sessions:**

- New York: The Wonder City (7/1)
- Secret Friend (7/12)

**Coming Sessions:**

- Covenant (7/24)
- Lie to Me (TBD)
- New Session (Date)

Figure 15: Shows a return to the notion of an 'Arc' (last seen in Error! Reference source not found.), with a dramatically simplified interface, and a new unit of story; the 'Session'.

---

44 For the purposes of this design section, assume that Figure 4 on page 6 transitions directly to image #5 of Figure 6 on page 7.

45 One significant feature of these early designs that has not explicitly been retained in later iterations of the design is the ability to edit multiple 'stories' at once. This fell to the wayside initially because it was better suited to a content designer tool than a player tool; most players would not be editing content in bulk. More pertinently, in later iterations of the design I discovered that it would actually be counter-productive to the 'session' model of play that I want to model and encourage.
To that end, I returned to the essential unit of face-to-face tabletop role-playing games: the 'session'. A loosely planned, largely improvised sequence of events, negotiated between GM and player. This new data structure solved two significant information hierarchy issues: there was now a clear location to store information logged from a period of play, and an easy storage system for events which would allow GMs to 'plan' to use a scene, making that available for use during a 'session' of play, while not locking them into a rigid timetable for those events to occur within. Since, unlike sessions as a whole, scenes do tend to be far more deliberately planned and set into motion; MAST could be appropriately complex where that complexity was desired, without overloading the GM with pre-planning tools.

46 One significant feature of these early designs that has not explicitly been retained in later iterations of the design is the ability to edit multiple 'stories' at once. This fell to the wayside initially because it was better suited to a content designer tool than a player tool; most players would not be editing content in bulk. More pertinently, in later iterations of the design I discovered that it would actually be counter-productive to the 'session' model of play that I want to model and encourage.
This new design sees the removal of more complicated and conceptual spreads (which will see again later in character creation), in favor of a simple three-part structure: Context, Conflict, and Consequence (with somewhat evocative names for our purposes here). It does, however, retain some of the scripting elements associated with earlier versions, in scaled down forms. Context refers to the narrative backdrop to a significant event; the undercurrents, motivations, and previous resolutions that have brought the characters to where they are. Conflict is the drama around which the arc turns; the decision the characters will need to face, or the problem they will have to resolve. Consequence is not just the outcome of the conflict, but the ramifications of that conflict on arcs further down the line.

Figure 17: The Context screen, showing the contextual interactions within an arc. Here GMs can transcribe notes, classify scenes (allowing them quicker sorting in the session planning portion of the tool), and selectively push information to the Player Handbook. Not pictured: Toggle Player View, a screen which would allow GMs to see content as it would appear in the Player Handbook, a necessary feature if we for any selective scripting (as shown in Figure 25, on page 67).
Figure 18: The Conflict screen, showing the conflict-centered interactions within an arc. Here GMs can still transcribe notes and select scenes, and can now set objectives, allowing them to explicitly direct players and clarify goals where needed. Players, in the Player Handbook, can annotate this list, which will be seen by the GM in the Toggle Player View screen.
Figure 19: The Consequence screen, showing the consequential interactions within an arc. Here GMs can still transcribe notes and select scenes, and can now set rewards and change the status of NPCs (de-spawning NPCs which will no longer be used, or marking an NPC as 'injured' to reflect the result of a conflict).

The Context, Conflict, and Consequence screens allow players to make high level decisions about the kinds of scenes they want to create, and pre-sort them by those kinds, for ease of access when planning a specific section. Scenes, in the context of a tabletop role-playing game can encompass something as specific as a single conversation between the players and an NPC, or as broad as scavenging for information from various locations and NPCs that can eventually be used to confront a different NPC.
Figure 20: Two views of the 'Library' (reached by clicking on the button, anywhere a tarot card has been or could be placed.

For GMs to gain a clear understanding of how the metaphors of tarot cards work, and treat them as a constructive piece of information, it needs to be easy to gain information about what a card means, and
how that card might be used. The library, as a result, can be reached from any slot where a tarot card can be placed.
Chapter 3: Artifact Design

MAST (Metaphoric Adventure-Scripting Tool): A user facing tool for creating multiplayer MMO narratives  Audrey Whitman

Figure 21: A 'Previous' session and a 'Coming' session, showing the interactions within a session.

The session screens, as discussed previously, allow players to select from scenes which they have already created, to loosely plan a period of play around them.
Figure 22: The Scene Creation and Editing screen, showing an existing scene, and the events it contains. Events can be typed according to 'Story Beats' and 'Resolutions'. 'NPC Changes' provides a list of all modifications made to a character or NPC during the course of a scene.

A story beat is a unit of a scene in a story where there is an exchange between the characters and the action (or reaction or revelation) advances the story, and shapes the development of that scene. A Resolution is a concluding event, which ends a scene. An NPC Change is a collation feature which GMs can use to visualize the progression of a scene.
Figure 23: Two scene editing screens; Edit Story Beat Event and New Resolution Event. These screens show a point and click interface for creating simple events very quickly. The condition statement created is displayed for the GM to view as parameters are added.

Through these tiny actions, GMs can create both generic events ('A monster appears', etc) and events with specific narrative or mechanical functions. As shown in the rightmost screen, an NPC by the name of Luca Zhiganov has been added to the 'God Knows I Had Plans' resolution event. When this event is triggered by the GM, the NPC will spawn (at a location determined on execution). The GM can then assume control of that NPC to talk with the players and deliver an item spawned with that NPC.
Figure 24: Two NPC creation screens, showing the ways a GM can create characters to populate a story. The rightmost screen shows the abilities an NPC will have when a GM chooses to ‘run’ them in combat. Players will have a similar interface in the Player Handbook for creating their own character-narrative spreads, which can then be viewed by GMs (at the Arc and Story level).

GMs can, at the Arc and Story level, create NPCs which are then available for use at the Arc, Scene, and Event level. Because NPCs are part of both stories and arcs, they can be modified by resolution states; so
we need to have the capacity to set similar conditionals in NPC description or informational text. For example, an NPC bio might look like the following:

"Little timmy looks up at you with big blue eyes [IF timmys_friend: 'which brighten at your approach'; ELSE: ', and he cowers at your approach']. He says 'What can I do for you, mate?'"

Figure 25: Shows an example of a conditional statement in a text description; borrows from the same principle in interactive fiction editors like Inform.

**GM and Player Tools Design**

While simplifying the scripting process relieves some of the tedium of content creation, it isn't a complete solution. 'Pre-Session' scripting is still time spent outside the game proper, preparing for someone else's interactive experience rather than being itself interactive. And while (as we have seen) that process can be streamlined and made more useful for the experienced or novice GM, it doesn't produce a more interactive experience for those GMs. How can we design a corresponding tool to MAST which provides GMs with the interactive part of the group storytelling experience?

Here we need to take a step back and consider the relationship between GMs and players in traditional table-top role playing games. What do GMs do, and what makes GMing fun for them? A GM typically performs the following functions in a given session:

- Describe Locations/Actions
- Talk as an NPC
- Direct Player Attention
- Trigger Events to shape/balance a session of play (including):
  - creating 'skill' challenges
  - posing 'logic' challenges
mediating ‘conflict’ challenges (both combat & argument)

playing ‘plot’ scenes

Log events as notes for next session

These functions broadly fall into ‘orchestral’ and ‘adversarial’ roles; directing players and providing challenges for them to face. The ‘fun’ of GMing, so to speak, is balancing these roles; providing challenge, direction, and character to a story told through multiple players. A tool which truly encourages the spirit of traditional table-top role-playing in a digital context needs to provide those same orchestral and adversarial abilities. To that end, MAST needs an in-game counterpart; a suite of GM ‘Tools’ that would help a GM to engage in the same type of synchronous interactive experience found in table-top role-playing, rather than the asynchronous experience of a content designer.

Features

Broadly, the GM Tools should integrate with both MAST and 5 Boroughs, insofar as it will be both drawing from and writing data to both. More specifically, the GM Tools must come pre-loaded with ‘generic’ encounter types. These templates will provide both examples for experimentation, and shortcuts for ‘drop-in’ use. As part of the MAST integration, the GM Tools must have access to types of ‘events’ and ‘NPCs’ that the player creates, filtered according to player-assigned ‘type’ and ‘story’. The interface tool itself should ‘turn on’ when a player who owns a ‘story’ begins that story with a group of designated players. Players are invited into a story (which essentially takes place in a private instance of 5 Boroughs) by the owner of that story, and cannot trade their membership to one another. In order to better sync up an ‘online’ gaming group to a ‘traditional’ friends-based gaming group, the GM Tools/MAST should hook into something that transmits data to appropriate places (in-game player apartments, social network feeds, etc).
Given the orchestral and adversarial roles described previously, what would these GM Tools need to be able to do? This should be approached as an interface overlay; if the players are playing a multi/single-player game, the GM is playing a tactics game

**Allow simple NPC 'ghosting'**

This allows GMs to quickly and simply 'take control' of a given NPC, speaking through them. This both maintains a sense of dramatic consistency, and provides clear dialogue logs for players and GMs to read through after the fact. More importantly, it frees GMs from needing to pre-write significant quantities of navigable dialogue for players to page through while the GM watches. Each NPC to PC engagement occurs in real-time, and involves both the player and the GM.
Figure 26: (Left) Player View of 5 Boroughs; (Right) GM View of a Group Role-Playing Session in 5 Boroughs.
Allow real-time placement of pre-created NPCs

This allows GMs to place NPCs created in MAST where and when they are needed, according to the flow of the story. Rather than setting pre-determined 'spawn points' or creating 'paths' that an NPC will walk along, a GM can place an NPC where the narrative calls for them to be. This frees MAST from needing to create a 'clockwork universe' where GMs must pre-plan a fully-operational world which anticipates the behavior of players; and frees GMs to work both before and during a session of play to create a sense of mood and dramatic climax.

Figure 27: the 'Place NPC' Icon selected, the placement map pops up.
Figure 28: When the 'Your NPCs' Icon is selected, the player creates a list by making the initial selections shown in the image to the left. Then the NPC list will pops up, with 'on hover' information about a given NPC.

Figure 29: When an NPC is selected, the player can drag an icon to place that NPC on the map. Clicking 'Place NPC' will spawn that NPC at the designated location.

Allow real-time triggering of pre-created 'events'

Some GMs will want to create scenes or 'events' that will play out when the players reach a specific place, or complete an action; and MAST will let them script those events. GMs may also want to create 'types' of events which can occur at any time. The GM Tools, in turn, must let GMs trigger these events manually, based on player behavior. This allows GMs to control the flow of a session of play (keeping all players active and engaged) and the speed at which the narrative progresses.

Allow top down combat/conflict control (run the actions of multiple NPCs)

This will allow GMs to control larger conflicts (many to many combat or conflict encounters) cleanly and with a minimum of confusion. A streamlined interface here will make it significantly easier for GMs to include combat or conflict challenges in their stories.
Create a log of actions, challenge results, dialogue, & 'event' IDs

This will allow both GMs and players to refer back to events that occurred in previous sessions; in order to analyze those events and better plan for subsequent sessions. For GMs specifically, it will provide significant statistical data on how well players are performing at the challenges the GM is setting for them, and the degree to which they are requiring prompting (which will help GMs to better tune future sessions).
Final Design Iteration for the GM Tools

Based on the feature list described above, we’ll briefly walk through that functionality as it might appear in an integrated tool which works with MAST and the Player Tools (as shown in the data flow chart in Chapter 3, Architectural Design, page 53). The GM Tools directly overlay onto the game (as shown in later screenshots); as a launch-able service within 5 Boroughs. While this results in some duplicated content (several screens are nearly identical between MAST and the GM Tools), it allows players to easily access their pre-planning materials within the game itself (rather than forcing them to keep MAST and 5 Boroughs open at the same time).
Choose a Story:

Minimal

The Mary Onettes

Arcs:

Sisters of Mercy

Arc Information

Dramatis Personae

Nancy (Mafioso)

Eddy (Politician)

Missy (Artist)

Description

This arc is about terrible references to band names, science, and blood-sports.

[Click to launch MAST and edit]

Broder Daniel

Saxon Shore

Figure 32: The entry point screens, showing the story selection process, choosing a story arc, and viewing the sessions associated with that arc. The functionality and the design of these screens mirrors that shown in [REF], on page [REF] in Chapter 3.
Chapter 3: Artifact Design

MAST (Metaphoric Adventure-Scripting Tool): A user facing tool for creating multiplayer MMO narratives

Audrey Whitman

Figure 33: Two screens associated with viewing session information in-game. The functionality and the design of these screens mirrors that shown in [REF], on page [REF] in Chapter 3.

The prior screens, as mentioned, deliberately duplicate the appearance and the navigational functionality of several screens in MAST. However, as shown above, this is as far as the resemblance goes; players cannot edit sessions or scenes, or add players. They must launch MAST to make any significant changes. For practical reasons (a GM must be logged into 5 Boroughs to lead a role-playing group), a new session can only be started from within the GM Tools.

The following images show how the in-session GM Tools might look, and the sort of functionality they would offer.
Figure 34: The Scene & Trigger Controls. Here we can see both how a GM might trigger a scene, and how an event which is being triggered automatically could be seen by the GM. Also included in these tools are manually-delivered rewards and items. Things which the GM chooses to deliver to the player directly, rather than attaching them to an NPC.

The Scene & Trigger Controls cover everything a GM might need to advance the plot of a session, or to reward and direct players. Like earlier screens, no new items can be created in the GM Tools; GMs simply have access to items, events, etc which they have already created through MAST.
Chapter 3: Artifact Design

MAST (Metaphoric Adventure-Scripting Tool): A user facing tool for creating multiplayer MMO narratives  Audrey Whitman

Figure 35: The Combat Controls. Here we see how a GM might control an NPC in combat, how they might spawn new NPCs as needed (and place them), and how combat can be started and stopped.

The Combat Controls cover everything a GM might need to orchestrate and participate in a combat encounter (both with and against) the players. The NPCs which appear in the Spawn New NPC option box here reflect the selections made during NPC Character creation (see Figure 24 on page 66). In other words, an NPC with only the ‘Combat’ checkbox clicked will appear in this list, but an NPC with only the ‘Conversation’ checkbox clicked will not.

Figure 36: The NPC Controls. Here we see how a GM might assume control of a specific NPC (or a group of NPCs), and the types of interactions they might be able to perform as an NPC.

The NPC Controls duplicate some functionality of the Combat Controls (spawning, specifically); primarily to prevent GMs from needed to jump back and forth between the two ‘modes’ during combat encounters. As above, the NPCs which appear in the Spawn New NPC option box here reflect the selections made during NPC Character creation (see Figure 24 on page 66).
Final Design Iteration for the Player Tools

The Player Tools, while a part of the data machinery shown in the data flow chart (in Chapter 3, Architectural Design, page 53) primarily receive data from the other two tools, and as such, primarily collate and represent that data. The following two screenshots provide an example of how information GMs release from MAST will appear to players.

Since the player tools are primarily representational, there is less functionality to document. Further demonstration of how specific data will appear can be found in Chapter 5, which will track a story from creation in MAST to execution in the GM and Player Tools.
Conclusion

While these tools obviously need further exploration, we can see here the beginning an approach to their design. By focusing on the needs of both players and GMs, we can better produce a set of tools which both meets the expectations set by offline play, and improves upon those experiences in meaningful ways. With these structural supports in place, role-playing in online spaces can benefit from improved immersion and player interaction, rather than be relegated to a necessity born out of distance, inevitably inferior to the 'real thing'.
Chapter 4: Walkthrough

This section will contain a complete walkthrough, detailing the process of using MAST from creating a new story, through the execution of that story using the GM Tools, and the experience of that story in the Player Tools.

Script

Take, for example, the following script. We have a story: The Case of the Disappearing Heiress; with an arc: Even the Least of Us. This arc has the goals: Find Timmie Tenspot's Sister, & Deliver his sister from poverty. The first goal contains the following scenes:

- Meeting Timmie
- Searching for Clues
- Spying on the Orphanage
- A Daring Escape
- Capture & Deflection
- Trust me, I'm a Doctor
- A Joyful Reunion

The first scene, Meeting Timmie, has the following events:

- How Can I Trust You?
- Music by Gaslight
- The Music Box
- Help Me Find Her

The second goal contains the following scenes:

- Writing a Letter
- Have You Seen This Man?
- Brother Can You Spare a Dime?
- Honest Day's Work
- Annie Finds a Home, Loses a Brother
- Can You Take a Letter?
- Better That She Doesn't Know

In Figure 38, on the following page, we can see the hierarchy of information contained in this narrative structure.
Figure 38: An example of MAST’s narrative encapsulation.

Screen By Screen

MAST

The following four screen (Figure 39 through Figure 42) show the informational portions of the main level of navigation. The covers: what MAST is, what the Tarot is, how to interpret Tarot cards (broadly), and signing in to the application.

Figure 39: The main level of navigation, showing the sign in panel.
Chapter 4: GM and Player Tools Design

Figure 40: The main level of navigation, showing historical information about Tarot cards.

Figure 41: The main level of navigation, showing the basics of card interpretation.
Figure 42: The main level of navigation, showing a brief explanation of MAST.

The following two images (Figure 43 and Figure 44) show how a story is created. It is initially blank, and will slowly fill as the user adds information to the story.

Figure 43: The main level of navigation, showing how to start a new story.
Chapter 4: GM and Player Tools Design

MAST (Metaphoric Adventure-Scripting Tool): A user facing tool for creating multiplayer MMO narratives  Audrey Whitman

Figure 44: A newly created story, with no players or arcs. A description shows a green checkmark (approved as valid input by the database), and a green radio box showing that this information will be shared with players.

The following images (Figure 45 through Figure 52) show the process of adding Players and NPCs to a story, and creating NPCs.

Figure 45: Adding a new player, entering their in-game screen name, their character name, and the character class they have chosen. This information can also be delivered to a story from the Player Tools. The green checkmark (content approved as valid input by the database) convention continues.
Chapter 4: GM and Player Tools Design

**MAST**

*The Case of the Disappearing Heiress*

Players:

Slepair O’Hara (Artist)

**Add Player**

- **Player Name**
- **Character Name**
- **Character Class**

*Add NPC*

**Figure 46:** A new character, once added, can be removed by clicking on the red symbol.

**MAST**

*The Case of the Disappearing Heiress*

Players:

Slepair O’Hara (Artist)

**Add Player**

*Add NPC*

- **Character Name** Wanda Lovelace
- **Character Class** Artist

*Create NPC*

**Figure 47:** The first screen of creating a new NPC.
Figure 48: The second screen of NPC creation, showing the 'alignment qualities' associated with characters in *5 Boroughs*, represented in 'Affiliations'. Also shown are the selection tools for determining the kind of actions an NPC character can take, and the kind of character reading you will perform.
Figure 49: The second screen of NPC creation, showing the drawing of the deck - the first stage of constructing a character reading.
Figure 50: The second screen of NPC creation, showing card selection in a character reading.
Figure 51: The second screen of NPC creation, showing a complete character reading, with generated interpretive information about the character, and sample ‘motivations’ which the Storyteller can use to define the behavior of that character in game.
### Chapter 4: GM and Player Tools Design

**MAST (Metaphoric Adventure-Scripting Tool):** A user-facing tool for creating multiplayer MMO narratives  
**Audrey Whitman**

#### Figure 52: The final screen of character creation, showing the statistics block associated with 5 Boroughs, a list of available physical and or social combat abilities, and any items the NPC might be carrying.

The following images (Figure 53 through Figure 57) show the process of creating a new arc, and how the card library can be used as a reference.

#### Figure 53: The first screen of creating a new Arc.
Dramatically, each arc must have a context of action, a primary conflict, and a resolution for that conflict. To encourage developers to follow this model, the arc is composed around these cards, which are used to influence the events associated with them.

Figure 54: The second screen of creating a new arc; demonstrating the Context, Conflict, Consequence narrative device. Dramatically, each arc must have a context of action, a primary conflict, and a resolution for that conflict. To encourage developers to follow this model, the arc is composed around these cards, which are used to influence the events associated with them.
Figure 55: The second screen of creating a new arc; demonstrating how to access the library.
Usually we think of strength in physical terms - big arms, powerful legs - but there is also inner strength. Inner strength comes from an exercise of the heart muscle. It is perseverance, courage, resolve and composure - qualities that help us endure when times are tough. In the past, a person with inner strength was commonly said to have character; he or she could be counted on in the darkest moments. Card 8 represents this energy of quiet determination. Strength is not a flashy card, but one that is solid and reliable.

**Figure 56:** The library, showing a variety of information about the selected card; including major themes, possible uses, and template events which can be attached to a scene in an arc.
Figure 57: The full navigable library, clicking on a card here will bring up the display shown in Figure 56 on the previous page.

The following screens (Figure 58 through Figure 60) show how to add a scene to an arc, and add events to scenes.
Add Arc

Edit Arcs
Absent Friends
Even the Least of Us

Arc Goals
1) Find Tinnies and Tepper's sister
2) Deliver key item from prison

Players
- Deputy Owne (Detective)
- Marty Delan (Politician)
- Germane Grey (Lawyer)

NPCs
- Wanda Lovelace (Actress)
- Iona McBeth (Believer)

Select a card:

Scene

Context
7 of Swords
Position
The situation which prevails
meaning
Running away, the 'Zone Wolf', hidden
Themes
Deceit, brains, secret plans, losing society

Consequence

Songbird

Save Arc

Figure 58: Creating a scene; demonstrating interface conventions mentioned throughout.
Figure 59: Creating an event; showing generated 'Objectives' which will be delivered to players as hints or tasks to guide their progress through the world. They are editable.

To create an event, a storyteller combines conditions to form an If/Then statement, shown in the bottom of the Events panel as it is being generated. Players can add When, Where, What, and How\(^\text{47}\) as condition types, which modify the circumstances under which an effect is triggered.

\(^{47}\) 'How' can only take a 'What' conditional phrase as it's argument.
Figure 60: An alternate view of Figure 59, showing a variety of the types of conditions which can be created.

The following screens (Figure 61 and Figure 62) show how a session can be created and filled with content.
Chapter 4: GM and Player Tools Design

MAST (Metaphoric Adventure-Scripting Tool): A user facing tool for creating multiplayer MMO narratives  Audrey Whitman

Figure 62: The second screen of session creation.

GM Tools

The following images (Figure 63 and Figure 64), show entering the Storyteller Tools, and selecting the storyline created in MAST.

Figure 63: Navigating through current, old, and planned stories.
Figure 64: Selecting a specific session which has not been run, and launching that session from the Storyteller Tools into MAST.

The following screens (Figure 65 through Figure 73) show the in-game controls, allowing a Storyteller to: trigger an event, allow an event to be triggered, deliver stuff as a Storyteller, spawn an NPC, talk, perform an action, & deliver an object as an NPC, start combat, control an NPC during combat, end combat, and end a session.

Figure 65: The in-game Storyteller interface, showing the tools they will have access to, and inviting the planned players.
Players

Notes

In the scene, I want to experiment with starting a scene in Multimedia so we’re going to begin a “flow” into the story, and “flashback” to later parts of the story as I like.

There’s a distracting week, seeing the scene of Wand’s murdered apartment. To get him to share what he knows, players must help him locate the woman’s sister where her 13-year-old cousin lives. For the family member who Wand is supposed to be too proud to accept help from.

Current Location: County Courthouse

Deliver Items

Select a Scene:

The Case of the Disappearing Heiress

Figure 66: Scene Controls - triggering a scene.

Players

Notes

In the scene, I want to experiment with starting a scene in Multimedia so we’re going to begin a “flow” into the story, and “flashback” to later parts of the story as I like.

There’s a distracting week, seeing the scene of Wand’s murdered apartment. To get him to share what he knows, players must help him locate the woman’s sister where her 13-year-old cousin lives. For the family member who Wand is supposed to be too proud to accept help from.

Current Location: County Courthouse

Select an Item: Delia’s Diary

Deliver to: Delia’s Diary

The Case of the Disappearing Heiress

Figure 67: Scene Controls - delivering an item.
Chapter 4: GM and Player Tools Design

MAST (Metaphoric Adventure-Scripting Tool): A user facing tool for creating multiplayer MMO narratives

Audrey Whitman

Figure 68: NPC Controls- selecting an NPC.

Figure 69: NPC Controls- placing an NPC.
Chapter 4: GM and Player Tools Design

MAST (Metaphoric Adventure-Scripting Tool): A user facing tool for creating multiplayer MMO narratives

Audrey Whitman

Players

Notes

In this session, I want to experiment with starting a game in MobaLoan, so we're going to begin a flow into the story, and 'Flashback' to later parts of the story as I feel like.

Tuners, in a desire to support a seen, seem fluffing the story of Wanda's murdered apartment. To get him to share what he knows, players must help his found his younger sister and hide her from his own passage, eventually delivering him into the custody of a distant family member who Tuner is supposedly too proud to accept help from.

Figure 70: NPC Controls - interacting as an NPC.

Figure 71: Combat Controls - Controlling a character.
Chapter 4: GM and Player Tools Design

MAST (Metaphoric Adventure-Scripting Tool): A user facing tool for creating multiplayer MMO narratives

Audrey Whitman

Figure 72: Combat Controls- Starting and ending combat.

Figure 73: Ending a Session

The following screen, Figure 74, shows how the Storyteller can review the content of a completed session.
Chapter 4: GM and Player Tools Design

MAST (Metaphoric Adventure-Scripting Tool): A user facing tool for creating multiplayer MMO narratives

Audrey Whitman

Figure 74: Reviewing logs of gameplay for a completed session.

Player Tools

The following images (Figure 75 through Figure 81) demonstrate what the player can interact with outside of the context of a session, the Player Tools. This includes: entering a story, creating a character spread, editing generated ‘Motivations’, reviewing Story & Arc content delivered to players, viewing NPC & Location content delivered to players, and annotating information in the Player Tools.

Figure 75: Selecting a story.
Chapter 4: GM and Player Tools Design

**MAST (Metaphoric Adventure-Scripting Tool): A user facing tool for creating multiplayer MMO narratives**

**Audrey Whitman**

---

**Figure 76:** Reviewing story content, with player comments.

**Figure 77:** Reviewing location information.
Chapter 4: GM and Player Tools Design

Figure 78: Reviewing NPC information.

Figure 79: Reviewing player annotations on an item held in the group inventory.
Chapter 4: GM and Player Tools Design

MAST (Metaphoric Adventure-Scripting Tool): A user facing tool for creating multiplayer MMO narratives

Audrey Whitman

Figure 80: Reviewing previous sessions, and annotating scenes.

Figure 81: Reviewing the basic information of player-controlled NPCs.
Figure 82: Screen one of building and editing a player controlled NPC (or player character).

Figure 83: Screen two of building and editing a player controlled NPC (or player character).
Chapter 4: GM and Player Tools Design

The following images (Figure 84 and Figure 85) show: joining a session, interacting with NPCs and Players, and leaving a session.

Figure 84: Joining a group of players in a session of play.

Figure 85: Shows the interaction options available to players, and the method for leaving a session.
Chapter 5: Conclusion

MAST demonstrates one method of addressing two significant issues in massively multiplayer online games, particularly those of the 'themepark' or 'fixed synthetic' variety: content production and player agency. The traditional model of game content production and consumption fails to meet the needs of either designers or players. Players consume content far too quickly for content developers to keep up; leading to endemic content deficits, and subsequently, highly generic quests. This discourages players from engaging with what content is present in a quest, and contributes to an essentially mindless form of quest consumption; which results in content being consumed even more rapidly. As a result, game designers are struggling to 'patch' single-player content into multiplayer worlds by borrowing the storytelling tools of traditional media models; essentially forcing the player to 'play along' with the game, and pretend that they are alone in a multiplayer world.

A promising way to escape that vicious cycle is to find ways of encouraging collaborative narrative building among your players, which can still coexist with designer-led world building. When designers allow players to make 'real', but not necessarily 'official' content, it establishes a closer narrative relationship between the designer and the player, with storytellers willingly operating within the established tone and character of the world. Far more prosaically, this relationship also produces useful feedback on effective spatial design, and a higher visibility for the role-playing community. By being 'responsive' to the needs of that community (though not at the expense of other groups of players), the designers of that world can engage in mutually beneficial co-development of a multiplayer world. That co-development is relevant and necessary because continuing the single player content paradigm is clearly unsustainable. While it is currently the dominant paradigm, and produces an adequate narrative experience; it produces an experience which works against every natural affordance of a multiplayer
online world. This poor fit cheapens player interaction with the world, reducing it to tourism and a timesink for designers.

As a tool developed to meet this need for co-narration, MAST is an unusually responsive narrative tool for an otherwise ‘fixed synthetic’ world (as compared to City of Heroes & City of Villains), and is conceptually simpler than other visual story builders (as compared to Spore: Galactic Adventures). The combination of narrative freedom, and careful scaffolding (in the form of symbolic meaning and template content) provides an opportunity to show how user-produced content can improve both designer-produced content pipelines, and the relationship between designers and the role-playing community. Rather than performing the minutia of enemy placement, players will be able to focus on creating specific events and NPCs, and controlling the flow of a story over a session of play. GMs using MAST will essentially be creating set-pieces which sit on top of 5 Boroughs; through an interface which maintains the metaphor of the game itself, and provides a visual ‘dictionary’ of actions that are unique and evocative. That layer of interpretation, explicitly drawn from the tradition of co-narration and group storytelling which is built into the experience of using tarot cards, will provide an additional creative flexibility that I believe will be a significant draw for players.

Further Work

At this point in the design and development process, I would have three major goals: user testing, ethnographic research, and development. The walkthrough in Chapter 4 could function as either a complete click-through of MAST and the GM and Player Tools, or a paper prototype of its functionality. With that, I could begin performing basic user studies, allowing potential players to navigate the system and provide initial design feedback. Alongside that testing, I could return to the planned ethnographic study, which was dropped due to time constraints and lackluster recruitment. With more time, I could recruit more aggressively in-game, and gather feedback on the Mission Architect as it is used by players.
More time spent researching the usage of the Mission Architect in game would allow me to provide more accurate analysis on how a strong co-narrative tool would affect existing online role-playing forum communities; as well as how it would affect the development of the 5 Boroughs forum community.

Taking feedback from these sources, I could begin developing vertical slices of MAST (as well as the GM and Player Tools) for further iterations of user testing and development. Given the scope of the design, however, this would require either a significant development team, or a long development cycle. Were I to continue the development process, however, I would recommend developing MAST in AJAX, Flash, or Flex; and the GM/Player Tools in Python or Lua.
Citations and Additional Reading


Bishop, J. (2009). "Enhancing the understanding of genres of web-based communities: The role of the ecological cognition framework". International Journal of Web-Based Communities, 5(1), 4-17


Harðarson, Reynir. Themeparks and Sandboxes, IGDA Lecture. 2009


Mortensen, Torill. "WoW is the New MUD: Social Gaming from Text to Video". *Games and Culture*. 2006; 1: 397-413.


