Villains, Architects and Micro-Managers: What Tabula Rasa Teaches Us About Game Orchestration

T.C. Nicholas Graham[1], Irina Schumann[1,2], Mrunal Patel[1], Quentin Bellay[1], Raimund Dachselt[3]

[1]School of Computing, Queen's University, Kingston, ON, Canada
[2]Faculty of Computer Science, University of Magdeburg, Magdeburg, Germany
[3]Interactive Media Lab, Technische Universität Dresden, 01062 Dresden, Germany

{graham, mrunal}@cs.queensu.ca, schumannirina@aol.com, dachselt@acm.org

ABSTRACT

Players of digital games are limited by the constraints of the game's implementation. Players cannot fly a kite, plant a tree or make friends with a dragon if these activities were not coded within the game. Game orchestration relaxes these restrictions by allowing players to create game narratives and settings as the game is being played. This enables players to express their creativity beyond the strictures of the game's implementation. We present Tabula Rasa, a novel game orchestration tool based on an efficient tabletop interface. Based on a study of 20 game orchestration sessions using Tabula Rasa, we identify five behavioural patterns adopted by orchestrators, and four styles of collaborative interaction between orchestrators and players. Finally, we present recommendations for designers of game orchestration systems.

Author Keywords

Game orchestration, game design, tabletop gaming.

ACM Classification Keywords

H.5.3. Collaborative computing.

INTRODUCTION

In many traditional forms of gameplay, design is intermingled with play. Children create games on the fly, establishing the rules as they play [3]. Groups adapt games via house rules and handicaps [14]. In roleplaying games such as Dungeons and Dragons, a game master creates the game narrative for players through an interactive process of describing what the players see and responding to the players' actions [18]. In short, players feel free to invent and modify games as they play them.

In contrast, most digital games separate design and play. Developers ship games as static products. The creativity and flexibility of traditional games, where players fluidly switch between design and play, is simply not supported.

In this paper, we explore *game orchestration,* where a special class of player, the game orchestrator, is given the power to create and modify a digital game world at the same time as others are playing. We show how game orchestration enables creativity in gameplay, by supporting a real-time dialogue between designer and player.

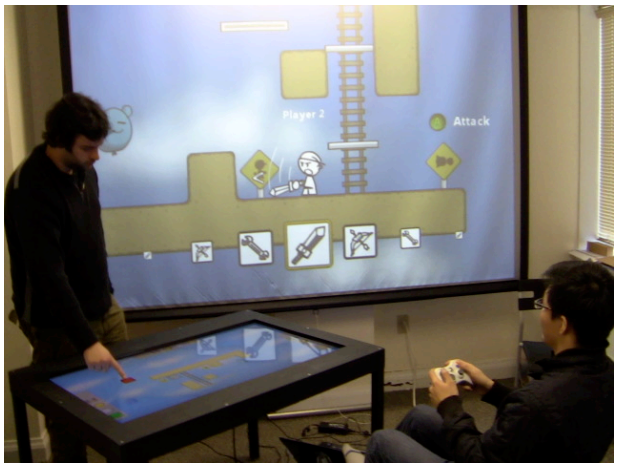

Figure 1: Tabula Rasa: a player (right) engages in a game created in real-time by an orchestrator (left)

To illustrate these concepts, we present Tabula Rasa, a tool for game orchestration (shown in Figure 1). Orchestrators use a digital tabletop to manipulate the game world in real time, using finger painting-like interactions. Orchestrators can change the game to adjust difficulty in response to players' frustration or boredom, and can create new parts of the world for players to explore. In sum, Tabula Rasa brings design-like activities to the runtime of digital games.

We have used Tabula Rasa to study how people use the ability to manipulate players' experiences at runtime. We were interested in questions such as: do orchestrators work quietly in the background, silently manipulating experiences, or do they verbally interact with players? Do orchestrators directly help players, or do they create situations and leave the players to succeed or fail on their own? Do orchestrators view themselves as collaborators, or as antagonists? In fact, we saw all of these behaviours and more, including the "helicopter parent" who rescues the player at the first hint of challenge, the "villain", who enjoys tormenting the player, and the "architect", who creates interesting locales for the player to discover.

CHI'13, April 27 – May 2, 2013, Paris, France.

The paper makes the following contributions: We show how game orchestration represents a new form of collaborative gameplay. We provide a comprehensive study of how people orchestrate games, in which we identify five behavioural patterns that orchestrators adopt, and four interaction styles within groups of orchestrators and players. Finally, we provide advice to designers of game orchestration systems.

BACKGROUND

"Orchestration" is the act of arranging or directing, often surreptitiously, to produce a desired effect [9]. The term *game orchestration* is therefore used to describe the creation and guiding of experiences for players as a game is being played [2, 7]. The notion of orchestrating games has existed in various forms over many years. Mulcaster described in 1581 the role of a referee in the game of "footeball" as a person who enforces rules and standards of play [6]. Since the mid 1970s, "game masters" or "dungeon masters" have been a critical component of pen-and-paper role-playing games such as Dungeons and Dragons [18]. A game master engages in a conversation with players, describing what the players see, playing the role of people that players encounter, and adjudicating combat. Thychsen et al. describe the role of the game master as maintaining the narrative flow of the game, enforcing rules, engaging the players, and managing the fictional environment [18].

More recently, game orchestration has been used in digital settings. Salovaara et al. provide a series of case studies of orchestration in digital games, identifying a range of roles including moderators, game masters, and people enabling infrastructure behind the scenes [15]. Crabtree et al. introduce the term "game orchestration" to describe the activities of operators of a mobile game based on SMS communication [2]. These operators interpret players' messages, tailor interaction with the player, pace the narrative flow, and guide players to effectively collaborate. Thus orchestrators work "behind the scenes", guiding players' experience. Neustaedter et al. use the term orchestration to describe the ongoing process of managing geocaching games [7]. Orchestrators engage in monitoring caches, players and non-players to ensure the game's smooth operation. As we see from these examples, game orchestration involves runtime manipulation of the game to guide the players' experience. We term the person carrying out these orchestration activities a "game orchestrator". (Note that this usage is unrelated to the body of literature on web service orchestration [11].) More concretely, we identify four ways in which games may be orchestrated:

Orchestrator as game master: This approach attempts to bring the experience of pen-and-paper roleplaying games (such as Dungeons and Dragons) to the digital world. Players gather around a digital table, which portrays maps and tokens for players in the game. A game master guides the story. The intimate feel of traditional roleplaying is preserved by the colocation of players around a table. Examples include TViews [5] and Wizard's Apprentice [9]. Game masters can be seen as a special kind of player, directing the game for their own enjoyment as well as that of the players. In Wizard's Apprentice, players can dynamically shift between orchestrator ("wizard") and player ("apprentice") roles.

Orchestrator as stage manager: Here, the orchestrator acts behind the scenes, manipulating the players' experience without their knowledge. Examples include the performers in the Desert Rain mixed reality performance [4] and Crabtree's "Day of the Figurines" described earlier [2]. Unlike the "orchestrator as game manager" approach described above, stage managers are hidden from the players, and roles are fixed – participants do not migrate between the player and stage manager roles.

Orchestrator as game designer: Orchestration can help in the game design process by allowing rapid sketching of game ideas before they are implemented in code [1]. An orchestrator uses a special interface to manipulate terrain and non-player characters, allowing design concepts to be quickly tested without the cost of implementing them. In the Raptor system, orchestrators use a digital tabletop surface to rapidly manipulate game content for players seated at a traditional PC and game controller [16].

Purposeful orchestration: Here, the orchestrator's goal is to create an experience that is helpful to its players. For example, in SIDES, an orchestrator guides the gameplay to provide therapeutic benefit to children with Asperger's syndrome [12]. In simulation-based training for military officers, orchestrators guide a training exercise behind the scenes, using a digital simulation environment to enact the role of personnel in the field [13].

A number of commercial digital games use orchestration techniques. Bioware's Neverwinter Nights provides a Dungeon Master interface, implementing a simple version of Dungeons and Dragons over the internet. Super Mario Brothers U for Nintendo's Wii U console features a limited version of orchestration, where one player using a touch tablet interface can help another player by modifying the world that she is traversing.

We distinguish game orchestration from the limited ability some games provide to players to modify the game world while playing. For example, in EA's Ultima Online, players can construct houses that others can visit, and in CCP's Eve Online, players can create missions for other players to carry out. In general, game mechanisms allowing players to direct the experience of other players differ over three axes:

Goal: In true orchestration, the goal is to create interesting experiences for the player. In contrast, for example, in Unknown Worlds' Natural Selection game, a "Commander" uses a special interface to provide move and attack orders to other players. For the Commander, the goal is to collaboratively win the game, and the players' enjoyment is incidental.

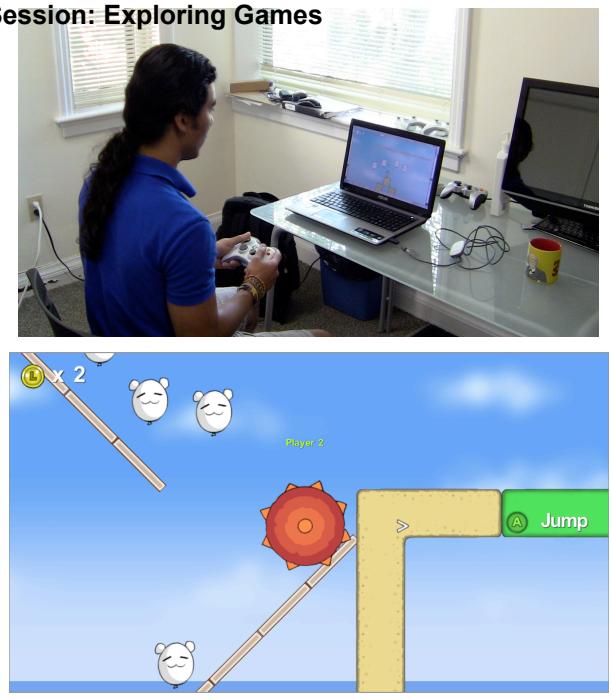

Figure 2: Player's view of Tabula Rasa. The player traverses a maze of platforms and ramps and collects balloons.

Interface: Orchestration systems typically provide a custom interface for orchestrators, as opposed to, for example, Ultima Online, where all players use the same interface.

Time: Although preparation may be performed in advance, orchestration is ultimately a play-time activity, as opposed to, for example, creating a modification ("mod") for the Half-Life 2 game for players to enjoy at a later time.

Despite that fact that game orchestration has been found useful and exciting in a number of contexts, there has been little study of how players use such a facility if available. This paper helps toward answering these questions.

TABULA RASA

To illustrate the concept of game orchestration, we created the Tabula Rasa game orchestration tool. The name comes from the Latin for "clean slate", indicating that the game can be arbitrarily created, modified and guided at runtime. Tabula Rasa takes place in a simple two-dimensional world where players use a standard Xbox 360 game controller to run, jump, pick up and use items (Figure 2). Tabula Rasa best supports action-based platform and puzzle games. Orchestration uses natural interaction, with a finger-painting style of interface for manipulating the game world. Tabula Rasa was implemented in C# using the Meerkat and XNA game development frameworks.

Game orchestrators guide the player's experience by creating and modifying the virtual world, adding or removing platforms and placing treasure in the world. An orchestrator uses a digital tabletop to rapidly create and modify terrain (Figure 1). Game players themselves do not use a tabletop, but play on a normal PC.

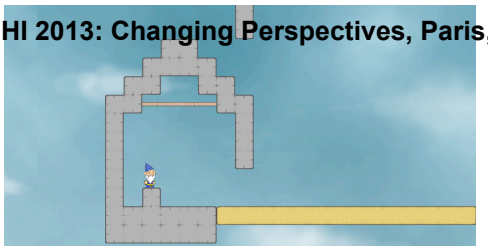

Figure 3: An orchestrator has created a simple house for a "wise old man"

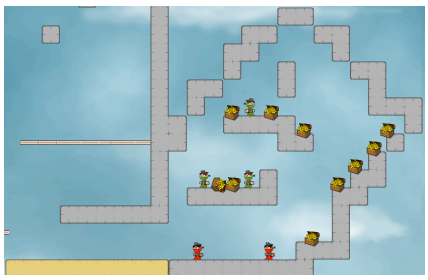

Figure 4: A treasure cave is heavily guarded by bandits

Player's Perspective

Figure 2 shows Tabula Rasa from a player's perspective. The player controls a spiky ball. A joystick on an Xbox 360 controller is used to move left and right. Pressing and holding the controller's "A" button causes the ball to jump. The player traverses a maze of ramps, platforms and jumping pads. Along the way, s/he can collect balloons for points. The challenge for the player is to find a way of traversing the complete level while collecting all balloons.

Orchestrator's Perspective

The orchestrator has a separate user interface dedicated to manipulating the game world itself. The orchestrator does not play the game, but modifies the game at runtime to enhance the player's experience. Specifically, the orchestrator can add and remove ramps, terrain blocks, balloons and non-player characters. If the modifications are in the player's field of view, the player sees them in real-time. The orchestrator may choose to make modifications off-screen to preserve the player's sense of immersion. By modifying the game world, the orchestrator can help the player past obstacles that are proving too difficult and can add completely new parts to the maze.

Figures 3 and 4 show examples of what an orchestrator can create using Tabula Rasa. Terrain blocks have been placed to create interesting structures. In Figure 3, a "wise old man" is located in a crudely drawn house. Figure 4 shows a cave containing bandits and treasure chests.

Orchestration Interface

The key to successful game orchestration is the provision of an interface allowing rapid creation and modification of the game world. If the player suffers lengthy waits while the orchestrator catches up, the illusion of the game can be broken. To support rapid orchestration, Tabula Rasa keeps the game world simple, so that it is easier to modify. As

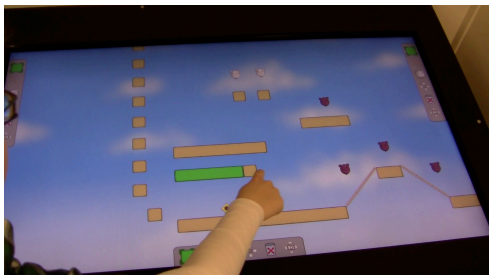

Figure 5: Finger-painting terrain

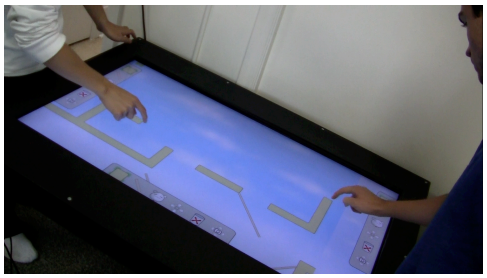

Figure 6: Two orchestrators working at the same time

shown in the earlier examples, the world is two-dimensional, based on a set of simple, pre-defined blocks. The orchestration tool is based on a finger-painting metaphor using a multi-touch digital tabletop. The orchestrator selects a drawing tool from a palette and drags her finger over the tabletop to create new terrain (Figure 5). The palette uses a set of drop-down menus to allow rapid selection of tools. Three palettes are arranged around the table, allowing groups of orchestrators to work together (Figure 6).

In addition to terrain, orchestrators can add objects to the world (e.g., a chest of gold), and non-player characters (NPCs). The world obeys standard physics, meaning that an object will fall to the floor when placed. This reduces the need for precise placement. NPCs can be moved by dragging out a path that the NPC will follow, allowing the orchestrator to quickly add life to a scene.

The use of a digital tabletop surface affords natural manipulation of the game scene. Orchestrators reported this interface to be enjoyable and intuitive, and to afford rapid manipulation of the game world.

This interface supports the forms of orchestration identified earlier. Orchestrators can act as "game masters", communicating with players while modifying the world, and dynamically changing between player and orchestrator roles by changing who is using the table. Or orchestrators can act as "stage managers", restricting players from seeing the table or actions they carry out. An orchestrator can restrict her actions to parts of the world that the player cannot see in order to reduce the players' awareness that the game is orchestrated. Orchestration can be collaboratively used by "game designers", where orchestrators and players sketch interesting play experiences that can later be coded into a standalone game. Because of this flexibility, Tabula Rasa makes an interesting platform for exploring how people use an orchestration system.

STUDY: GAME ORCHESTRATION WITH TABULA RASA

We used Tabula Rasa to explore how people play a game where orchestration is possible. We conducted an observational study to answer the following questions: How do players and orchestrators behave? What kinds of gameplay emerge? Can game orchestration be a fun experience for both orchestrator and player?

Equipment

The orchestrator used a 46" multi-touch display based on a PQ Labs G4 sensor. The display was mounted in a 30" high table, allowing comfortable use while seated or standing. The display resolution was 1920 x 1080 pixels. The player used a laptop computer with a 17" display operating at 1920 x 1080 pixels, and an Xbox 360 wireless game controller to control the game avatar.

The game world was partially populated with platforms, ramps and bouncing blocks. Large sections of the world were left empty. To give the game a goal, balloons were provided that, when popped, increased the player's score.

The player and orchestrator were in the same room. They were physically close enough to converse comfortably, but could not see the other's display. Two video cameras were used, recording the player and the orchestrator.

Recruiting

We recruited ten pairs of participants from the university community. Participants ranged from 18 to 40 years old, with a mean age of 25. Seven participants were female and 13 male. All 20 participants reported that they use a computer for at least two hours per day. All participants reported having played games on a multi-touch device.

Method

At the beginning of the session, the participants were read a letter of information describing the study, signed a consent form, and completed a demographic questionnaire. The concept of game orchestration was explained, and participants were introduced to the digital tabletop. The orchestration interface was demonstrated, and players were permitted to experiment with it until they were comfortable with its functionality.

One participant was randomly chosen to take on the orchestrator role, and the other acted as player. The participants were instructed to play for ten minutes. The only direction provided was that the orchestrator should create a fun experience for the player. Once the ten minutes had elapsed, participants completed a custom questionnaire reporting their experience. The participants then changed roles, and repeated the play session.

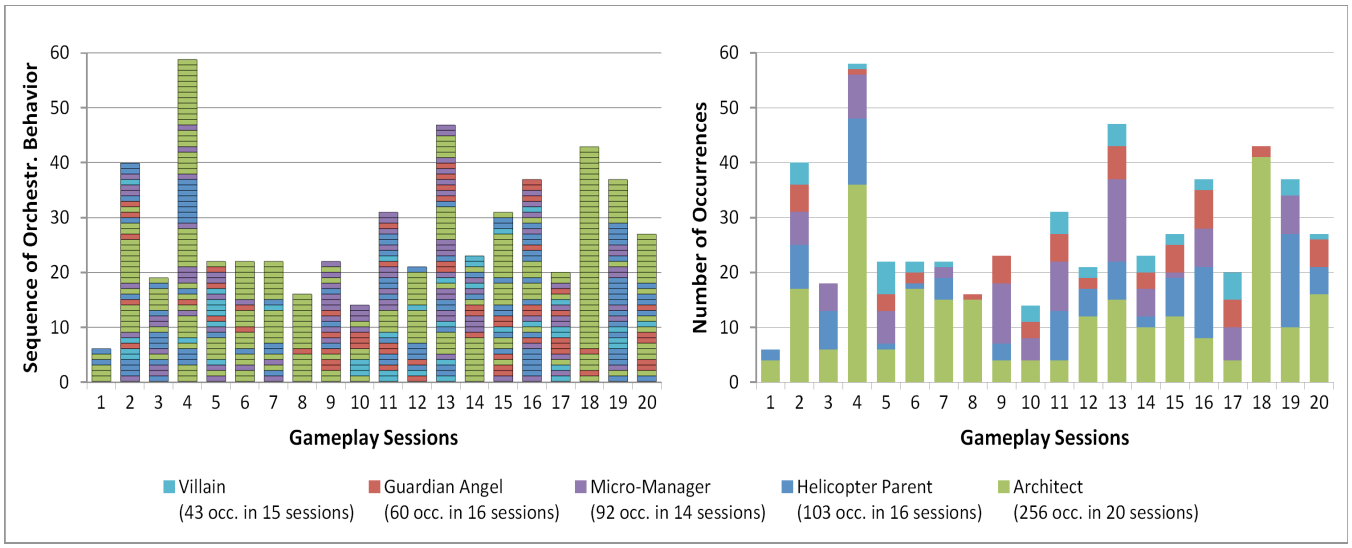

Figure 7: Types and distributions of orchestrator behaviours over all 20 sessions

Finally, a semi-structured interview was conducted to address interesting situations observed during the game.

Analysis

We used an open coding process [17] to identify behaviours and events of interest during the orchestration sessions. The videos for all 10 sessions were analyzed by two coders. Because participants acted as both player and orchestrator for 10 minutes each, there were 20 videos, for a total of 200 minutes. The player and orchestrator videos were viewed side by side to make it easy to see the interplay between the two participants, e.g., to see how a player reacted to an orchestrator's actions. Two meetings were held to review and adjust the list of codes and to resolve disagreements between the two coders. The codes were grouped into three categories: communication, orchestrator actions and player actions. For example, codes were identified which captured when an orchestrator built an area of the world that the player currently could not see, when an orchestrator modified the terrain to directly help a player, and when the orchestrator and player talked with each other. 25 codes were identified, and a total of 1,282 codes were assigned over the 200 minutes of video analyzed.

Groups of codes were identified as markers for interesting orchestrator behaviours and interesting styles of interaction between orchestrators and players. Each occurrence of an interesting behaviour or style was hand-verified by re-examining the videos.

RESULTS

The interviews and questionnaires showed that most orchestrators and players enjoyed their experience. From analysis of the videos, we identified five behavioural patterns that orchestrators adopt, and four styles of interaction between orchestrators and players.

Participant Experience

During the interview, orchestrators were asked whether they succeeded in creating a fun experience for the player. All 20 answered in the affirmative, indicating that orchestrators believed that Tabula Rasa allowed them to achieve the core goal of game orchestration. Participants' comments were highly positive, saying for example "I really enjoyed it" and "I love it". All 20 participants agreed that "It was easy to use the orchestration tool."

Impressions about the player role were positive, but not unanimously so. 18/20 participants agreed with the statement "I enjoyed playing the game while it was orchestrated" (responding 5 or higher on a 7-point Likert scale), and 10/20 strongly agreed with this statement (score of 7 on the 7-point scale). 19/20 agreed that "I would like to play games again that are manipulated by an orchestrator". This implies that most players found this an enjoyable experience, but that a minority did not. One of the two players reporting a negative experience complained that there were no concrete goals, and that the orchestrator's intervention removed any sense of challenge: "You have an infinite amount of coins. You have no goals. Talking with the orchestrator you can get anything you want, design everything that you want. There are no goals or anything to attain."

Orchestrators did not attempt to disguise their actions; all 20 participants agreed with the statement "The changes to the game by the orchestrator were very noticeable." Players largely appreciated the orchestrator's interventions, with 19 of 20 agreeing that "I liked that the orchestrator was there to help with difficult situations." However, a small minority (2/20) agreed that "I was annoyed because of the changes that were made by the orchestrator."

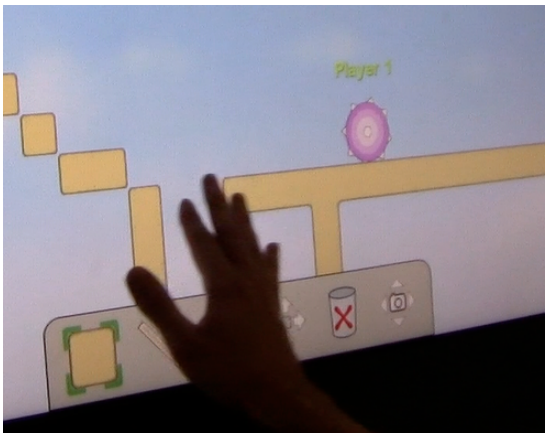

Figure 8: Helicopter Parenting: the orchestrator is closing off a pit before the player falls into it

Orchestrator Behavioural Patterns

During a session, orchestrators moved between a set of five behavioural patterns. These included the "helicopter parent", buzzing around the player to ensure he could make no errors, the "guardian angel" who kept the player out of trouble, the "architect", who worked independently of the player to create new and interesting terrains, the "micro-manager", who enforced his will at ever step, and the "villain", who aimed to spoil the player's fun. Each pattern was used by at least six orchestrators and was seen dozens of times over all sessions. Figure 7 summarizes the patterns that were seen in each of the 20 sessions. The left part of the figure shows the progression from one behaviour to the next, showing how frequently orchestrators changed their behaviour. The right part summarizes the number of occurrences of each behaviour.

Helicopter parent: This pattern involved "hovering" around players like a helicopter, ensuring that the player was never challenged by difficulties. If the player encountered a jump that required careful timing, the helicopter parent would quickly insert a ramp. If the player became stuck on complex terrain, the helicopter parent would immediately smooth the terrain to allow easy passage. We defined a helicopter parent as an orchestrator who would not permit players to ever fail, or even to experience the frustration of a challenge that needed to be resolved. An example of helicopter parenting is shown in Figure 8.

Helicopter parenting was observed frequently, with 16 of 20 orchestrators exhibiting this behaviour at some point in their session. For three of 20 orchestrators, this style dominated play, with more than ten instances observed throughout the ten-minute session.

Players' comments were mixed regarding helicopter parenting. Players described positively that this style of orchestration "makes it easier" and is "helpful" and that the orchestrator's intervention meant they "get to collect more points". Another complained that orchestration made the game "too easy".

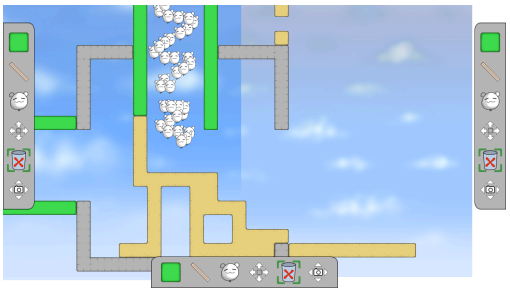

Figure 9: Pipe structure built by an architect

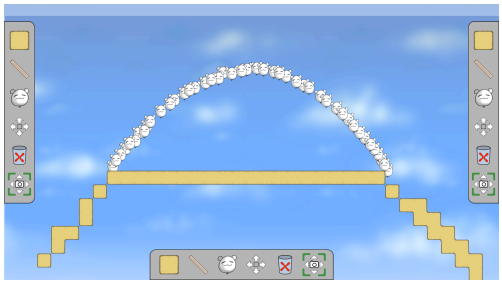

Figure 10: Arch-like structure built by an architect

Guardian angel: This type of orchestrator helped players when they encountered difficulty, but unlike the helicopter parent, permitted the player to struggle before intervening. In some cases, the guardian angel waited until the player clearly could not proceed; in others, the angel acted following a verbal request for help from the player.

In one example, a player fell into an impossibly deep pit. The guardian angel permitted the player to try several times to exit the pit, and only when it became clear that it was not possible to jump out did the orchestrator provide a ramp.

Guardian angel behaviour was frequently observed, with 16 of 20 orchestrators taking on this role at some point during their session.

Players' comments on the role of the guardian angel were positive, including that "it is nice to get help to reach places" and "it's nice to get help when you are stuck". One player commented that orchestration is "different from a computer manipulating the game; a person is more kind."

Architect: The architect operates in a part of the world that the player cannot see, creating interesting structures for the player to find and explore. Examples created by architects include a vertical pipe with bouncy walls filled with balloons (Figure 9) and a balloon arch (Figure 10). The architect pattern was seen in all 20 sessions. It was the most frequently adopted pattern, seen 256 times in total.

Architects operate surreptitiously, out of the player's view. This helps to provide the illusion that the world is static, not being modified under the player's feet.

Architects may suffer time pressure, as they need to complete their creations before the player arrives and starts interacting with them. Architects largely did not experience

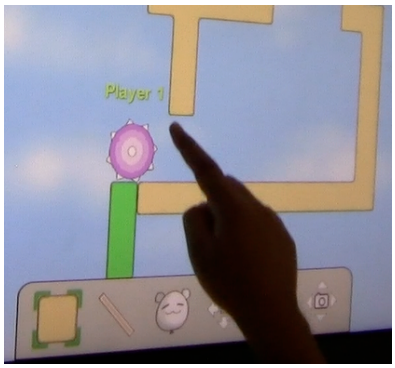

Figure 11: Micro-managing orchestrator draws a wall blocking the player's path, forcing her to go in the other direction

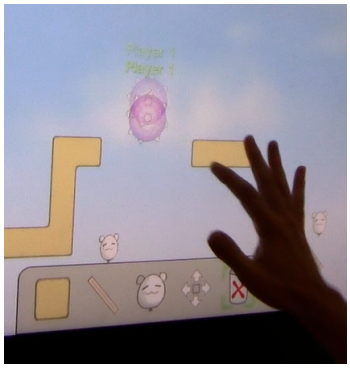

Figure 12: Villainous orchestrator removes the terrain underneath the player, causing him to fall

a sense of haste, however, with 18/20 agreeing that "I had enough time to do the changes I wanted to do."

Players responded positively to the actions of architects, specifying that it makes the world "more dynamic", and that the player "never knows what to expect". A minority found that architectural changes could be too far-reaching, for example saying that "changes were too quick; you need to go back and look", and "if there are too many changes, there is too much to adapt to".

Micro-manager: The micro-manager attempts to directly control the player's experience. For example, a micro-manager might wish a player to travel towards a new area that he has just created. The micro-manager may verbally instruct the player which way to go, or may even modify the terrain of the game world so that the player has no choice but to go in this direction.

In one session, the orchestrator found a tower of balloons on the left-hand side of the player. The orchestrator verbally instructed the player to keep jumping. While the player was jumping, the orchestrator inserted blocks underneath the player, and then directed her to free-fall right on the tower of balloons. Figure 11 shows the orchestrator drawing a wall in front of the player, giving her no choice but to move toward the tower of balloons.

Micro-management was seen in the majority of play sessions (14 of 20). Players' reactions were mixed. For example, one reported that "the orchestrator has too much power", and another "didn't know if the orchestrator helps or not". Micro-management was seen particularly frequently in two of 20 sessions, with more than ten occurrences in each. Surprisingly, in both cases, the players strongly agreed that it was enjoyable to play when the game was orchestrated, and strongly disagreed that orchestration was annoying. This implies that micro-management need not negatively affect the players' experience.

Villain: A villain attempts to spoil the player's fun. Examples of villainous behaviour include completely blocking the player with terrain so he cannot move, or removing the floor below a player just as he is about to complete a jump (Figure 12).

Villainous behaviour was common, seen in 15 of 20 sessions, but only two orchestrators repeatedly acted as a villain (6 times in one session and 5 in another). Following one such session, the player commented that orchestration is "bad if you are stuck with a mean orchestrator." Normally, however, orchestrators appear to have taken on the villainous role without malicious intent, as a prank or joke. Such jokes grow stale with repetition, explaining why the behaviour was rarely repeated.

Interaction Styles

Additionally, our observations revealed four distinct ways in which the player and the orchestrator interacted. These provide insights into the interpersonal dynamics involved in orchestrated gameplay. The interaction styles ranged from subtle manipulation behind the scenes, where the orchestrator attempted to keep her actions secret from the player, to full-on collaboration where the orchestrator and player worked together to achieve a goal in the game.

Orchestrator-dominant: By far the most frequent form of collaboration was "orchestrator dominant", where the orchestrator directs the progression of gameplay. The orchestrator may enforce this dominance either verbally or via the orchestration tools. This form of orchestration was seen in 19 of 20 sessions.

Orchestrators adopted both controlling and helpful forms of dominance. The micro-management of Figure 11 shows the controlling form. In a different session, the orchestrator helped a player who was unfamiliar with the use of a game controller, verbally directing him, and manipulating the terrain to ensure that he could progress.

Orchestrators commented that "it's fun that you can do whatever you want", and "[you're] playing god". One player characterized orchestrator dominance as an antagonistic relationship, where it is "the orchestrator versus the player". Another described the orchestrator as "helpful".

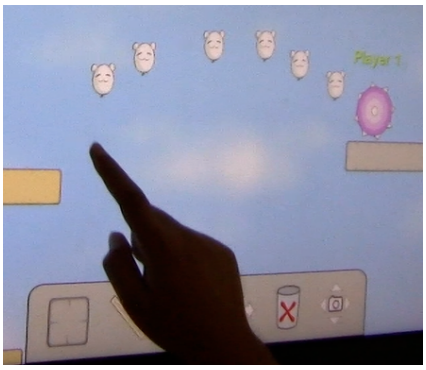

Figure 13: Orchestrator guiding the player with a trail of balloons

Guided: A second form of collaboration in which the orchestrator remains in control is "guided" interaction. Here, the orchestrator subtly directs the player where to go, in contrast to the heavy-handed approach of orchestrator dominance. In one example, the orchestrator wished to guide the player to a newly created part of the level. She did this using in-game cues, providing a trail of balloons leading in the correct direction (Figure 13). This gave the player a hint that there might be something interesting to find if the balloons were followed. Guides may spend significant time working "off screen", creating and modifying parts of the level that the player cannot see.

Guiding differs from orchestrator dominance in that the orchestrator does not directly insert herself into the game, but uses the game itself to guide the player's actions.

Guided orchestration was seen in 17/20 sessions, with a mean of 5 times per session.

Team: Here, the orchestrator and player establish a goal together, and verbally communicate to achieve this goal. Some examples of teamwork that were seen were:

- Climbing as high as possible (seen in two sessions), requiring the orchestrator to build platforms and ramps allowing the player to climb.
- Collaborating to get balloons that are out of the player's reach (seen three times). For example, an orchestrator and player collaboratively developed a plan to get the player over a high wall to retrieve a balloon on the other side.
- Collecting balloons in an acrobatic style (seen twice). The orchestrator and player collaborated on how to use bouncing blocks to allow a player to drop from a height and ricochet over an obstacle to collect a balloon.

Teamwork was the least frequently observed style, seen in 6/20 sessions. This was a particularly social activity, leading one participant to comment that orchestration is "fun depending on the kinds of friends you have".

Player-driven: Here, players dominate the collaboration, instructing the orchestrator what to do. This is distinguished from teamwork in that the player alone determines the goals, and that communication is limited to instructions from player to orchestrator.

Player-driven orchestration was seen in 10/20 sessions, and was used a mean of 4 times in sessions where it occurred. Typically, this form of orchestration arose when the player had a particular goal she wished to achieve, and enlisted the orchestrator's help. For example, one player wished to understand exactly how the game's physics worked. This player instructed the orchestrator where to build terrain in order to allow him to experiment with these mechanics.

Summary

We have described five orchestration behavioural patterns and four styles of interaction between orchestrator and player. All were derived from observation during our study.

It is important to note that orchestrators are not rigidly defined by a single pattern. On average, orchestrators changed type 27 times over their ten-minute session. Furthermore, on average, orchestrators took on four of the five types at some point during the session.

All 20 orchestrators acted as architects for a significant part of their session, implying that it was natural for them to create parts of the level off-screen so that players could encounter a completed design. Equally, most orchestrators also spent significant time closely helping or directing the player; 16/20 acted as helicopter parents, 16/20 acted as guardian angels, and 14/20 as micro-managers. Of these, the guardian angel was the least frequent (60 occurrences), while helicopter parent (103 occurrences) and micro-manager (92 occurrences) were more frequent. This implies that when directly observing players, orchestrators were inclined to significantly intervene in play, either by making sure they made no mistakes, or by forcing the player to play in the way the orchestrator has envisioned. It is interesting that relatively fewer orchestrators were content to let the player play uninterrupted.

Similarly, groups frequently changed their interaction style. On average, each session used 2.6 interaction styles, and groups changed style on average 20 times over the ten-minute session. Unsurprisingly, the dominant interaction styles were those where the orchestrator led. The orchestrator-dominant style was seen 276 times over 19/20 sessions. The guided style was seen 72 times over 17 sessions. The player-driven style was seen in half of the sessions, but with a much lower 29 occurrences. The team style was seen in six sessions, with a total of 24 occurrences. From this we conclude that orchestrators tend to control the session most of the time, but that many groups discovered other means of collaborating for a minority of the time.

This analysis helps in understanding why, as discussed earlier, one participant strongly agreed that "I was annoyed because of the changes that were made by the orchestrator."

The orchestrator in question exhibited six cases of micro-management, four instances of villainous behaviour, and 25 instances of orchestrator dominance, all within a ten-minute session – annoying behaviour indeed!

In sum, the study revealed a rich set of ways in which orchestrators create experiences for players, and varied ways in which the orchestrator and player interact.

ANALYSIS AND DISCUSSION

The study shows that orchestrators exhibit numerous behaviours, and frequently change between them. As was seen in Figure 7, some orchestrators tend to work quietly in the background (e.g., as architect), while others frequently intervene (as guardian angel or helicopter parent), often in a dominant or even villainous way. But in most cases, orchestrators follow all of these behavioural patterns over the course of a single session.

In our questionnaires and interviews, players did not complain about the intervention of orchestrators. We conclude that participants viewed orchestration as another form of playing the game, rather than as selflessly creating experiences for the player. During the frequent occurrences of orchestrator-dominant play and the occasions of villainous play, orchestrators appear to be valuing their own entertainment at least as highly as that of the players. Players also clearly did not expect the orchestrator to quietly remain in the background, as evidenced by the widespread occurrences of team and player-dominant interaction styles. The player viewed the orchestrator as a playmate to be engaged in a fun, collaborative activity. This differs considerably from the "stage manager" style of orchestration seen in, for example, Day of the Figurines [2], where orchestration is considered to be a strictly behind-the-scenes activity, and is closer to the game-master form of orchestration of pen-and-paper role-playing games.

We found that a digital tabletop surface has significant promise for game orchestration. The finger-painting approach allows smooth and fluid creation and deletion of terrain, and allows multiple orchestrators to modify the terrain at once. One major limitation that we have not explored is that the current system can provide only a few dozen tools before the palette becomes clumsy. We believe that this limitation is not critical, however, as interesting terrains and stories can be created even with only a small palette. Additionally, predefined palettes could easily be selected based on the kind of gameplay that is anticipated (e.g., an exterior fantasy setting or a modern building interior), or even switched at runtime. While large tabletop displays are not yet common, many forms of touch-sensitive surfaces have become available at consumer prices, including tablet computers with up to 27" surfaces.

Another limitation of our approach is the restriction of games to two dimensions. The earlier Raptor system [16] has shown that it is possible to sketch 3D games using a 2D top-down view, making it possible to orchestrate, for example, a 3D racing game. This top-down approach is also used in some commercial tools, notably the Neverwinter Nights Aurora Engine. A more general approach would be to use a 3D editor such as in commercial tools like Autodesk Maya and 3DS Max. These full-featured tools, designed for professional artists, do not support the speed of editing required for orchestration. Simplified 3D editing tools can reduce expressiveness in favour of simplicity and fast editing [7]. There is room for study of the tradeoffs involved in these different approaches for presenting the game world to the orchestrator.

Our study picked one of many possible group structures, with only one orchestrator and one player, co-located in the same room. Anecdotally, we have discovered that the positioning of the participants can have a significant effect on the interaction style.

LESSONS FOR GAME DESIGNERS

Our study has shown that game orchestration is a novel and interesting way of bringing design-like activities to runtime. The existence of commercial games like Neverwinter Nights and Super Mario Brothers U implies that game orchestration is an exciting way to provide novel and creative gameplay. Drawing from our study, we therefore propose specific lessons for designers of games incorporating orchestration features:

- *Don't design for a single behavioural pattern.* As we have seen, orchestrators change behaviour frequently and fluidly over the course of the orchestration session. Game orchestration systems should permit easy adoption of the full range of patterns.
- *Support flexible asymmetry.* In a minority of our sessions, villainous or micro-managing orchestrators created a negative experience for players. It should be easy for participants to change who is player or orchestrator, so that no player is held hostage by a malevolent orchestrator. In Tabula Rasa, this is as easy as changing who is sitting at the table and who is holding the game controller.
- *Support speed over expressiveness.* It is critical to be able to create new terrain and structures quickly, so that the player is not forced to wait. In Tabula Rasa, the use of a limited palette of blocky graphics and the use of a finger-painting style of interface allowed rapid manipulation of the world. No participants commented negatively about this limited palette, indicating that players did not view these crude graphics as overly limiting.
- *Consider orchestrators as players.* We saw strong evidence that orchestrators viewed themselves as a special kind of player. Rather than viewing their own entertainment as subsidiary to that of the player, they expected to be entertained in their own right. Game designers could consider ways of providing structured

goals in the game that will engage both orchestrator and player. For example, a game might include prizes for completing a level in a particular way – e.g., completing it quickly or using the fewest blocks.

CONCLUSION

In this paper, we have presented a study of game orchestration based on our Tabula Rasa tool. We have shown that when given game orchestration facilities, people take on a surprising range of behaviours. We identified five different behavioural patterns and observed that people frequently change between these behaviours over an orchestration session. We identified a range of collaboration styles between orchestrators and players, and found that these also change frequently during play sessions. Both orchestrators and players enjoyed this style of play. This study represents a first step in understanding the ways in which creative gameplay can be enabled by bringing design-like activities to a game's runtime. Our orchestrator patterns and interaction styles can now be used to creatively design novel games which take them into explicit consideration.

ACKNOWLEDGEMENTS

We would like to acknowledge the technical help provided by Ameer Hamza, Zi Ye and Christophe Bortolaso. This work was funded by the NSERC SurfNet Strategic Network and the LEIF Transatlantic Student Exchange Program. The paper benefited greatly from the reading of the reviewers.

REFERENCES

1. Agustin, M., Chuang, G., Delgado A., Ortega, A., Seaver, J. and Buchanan, J.W., Game sketching. In *Proc. Digital Interactive Media in Entertainment and Arts,* ACM Press (2007), 36-43.
2. Crabtree, A., Benford, S., Rodden, T., Capra, M., Flintham, M., Drozd, A., Tandavanitj, N., Adams, M. and Row Farr, J., The Cooperative Work of Gaming: Orchestrating a Mobile SMS Game. *Computer-Supported Cooperative Work* 16 (2007), 167-198.
3. Evaldsson, A.-C. and Corsaro, W.A., Play and Games in the Peer Cultures of Preschool and Preadolescent Children: An Interpretive Approach. *Childhood* 4:377 (2008), 377-402.
4. Koleva, B., Taylor, I., Benford, S., Fraser, M., Greenhalgh, C., Schnaedelbach, H., vom Lehn, D., Heath, C., Row-Farr, J. and Adams, M. Orchestrating a Mixed Reality Performance. In *Proc. Human Factors in Computing Systems,* ACM Press (2001), 38-45.
5. Mazalek, A., Mironer, B., O'Rear, E. and Van Devender, D. The TViews Table Role-Playing Game, *Journal of Virtual Reality and Broadcasting,* 5(2008):8, (2008).
6. Mulcaster, R., *Positions Wherein Those Primitive Circumstances Be Examined, Which Are Necessarie for the Training up of Children,* originally published 1581, available in reprinted form by Thoemmes Press (1995).
7. Neustaedter, C., Tang, A. and Judge, T.K., Creating scalable location-based games: lessons from Geocaching. Personal and Ubiquitous Computing (2011), 1-15.
8. Oh, J.-Y., Stuerzlinger, W. and Danahy, J., SESAME: Towards Better 3D Conceptual Design Systems, In *Proc. Designing Interactive Systems,* ACM Press (2006), 80-89.
9. *Oxford English Dictionary,* Third Edition, Sept. 2004.
10. Peitz, J., Björk, S. and Jäppinen, A. Wizard's Apprentice - Gameplay-Oriented Design of a Computer- Augmented Board Game. In *Proc. Advances in Computer Entertainment Technology,* ACM Press (2006), article 18.
11. Peltz, C., Web Services Orchestration and Choreography, *Computer* 36:10 (2003), 46-52.
12. Piper, A.M., O'Brien, E., Ringel Morris, M. and Winograd, T., SIDES: A Cooperative Tabletop Computer Game for Social Skills Development, In *Proc. Computer Supported Cooperative Work,* ACM Press (2006), 1–10.
13. Roman, P.A. and Brown, D., Games – Just How Serious Are They? In *Proc. Interservice/Industry Training, Simulation & Education Conference,* MetaPress (2008), article 8013.
14. Salen, K. and Zimmerman, E., Rules of Play: *Game Design Fundamentals.* MIT Press (2003).
15. Salovaara, A., Johnson, M., Toiskallio, K., Tiitta, S. and Turpeinen, M., Playmakers in Multiplayer Game Communities: their Importance and Motivations for Participation. In *Proc. Advances in Computer Entertainment Technology* (2005), 334-337.
16. Smith, J.D. and Graham, T.C.N. Raptor: Sketching Games with a Tabletop Computer. In *Proc. Futureplay,* ACM Press (2010), 191-198.
17. Strauss A.L. and Corbin J. *Basics of Qualitative Research: Techniques and Procedures, 2nd ed.* Sage (1998).
18. Tychsen, A., Hitchens, M., Brolund, T. and Kavakli, M., The Game Master. In *Proc. Australasian Conf. on Interactive Entertainment* (2005), 215-222.
19. Wong, N. and Gutwin, C., Controlling an Avatar's Pointing Gestures in Desktop Collaborative Virtual Environments. In *Proc. Group,* ACM Press (2012).