

The Tenth Annual Game Design Think Tank

Project Horseshoe 2015



Group Report: Of Minds and Mobs: Game Design for Shared Avatars and Other Weird Collectives

Participants: A.K.A. *"Body Politic"*

Daniel Cook, Spry Fox

Ted Castronova, Indiana University

Michelle Clough, freelance

Andrew Friedman

Tim Fowers

Facilitator: Tim Fowers

Brief statement of the problem(s) on which the group worked:

Most games have a unity between a single player and a single character. Our report looks at games that play with this assumption. We'll give examples where there's clearly a character or entity in the game, maybe a single-player character, but many players are trying to decide what that character will do. Or each player has some tiny role and the net effect of their play is some emergent thing, a mob. Games like Twitch plays Pokemon or Artemis suggest this is an exciting new design space that is only sparsely explored.

A brief statement of the group's solutions to those problems:

We looked at existing examples, identified key elements in the general playspace, and explored untapped spaces for shared avatar games. We codified different control schemes, mechanical vs mental unity, player motivations, and definitions of "shared avatar."

Introduction

Most games assign a single player to a single character. Our report looks at games that play with this assumption. There are **shared avatar** games like Twitch Player Pokemon where there is clearly a dominant character or entity in the game, but many players are trying to decide what that character will do. Or **crowd games** like Realm of the Mad God where each player has some tiny role and the net effect of their play is an emergent entity. Both are exciting new design spaces that designers have only begun to explore.

The team writing this paper is well versed in cooperative gameplay that lies at the heart of most Shared Avatar games. Tim Fowers has designed multiple cooperative board games (Burgle Bros and Wok Star). Daniel Cook also designs cooperative games (Realm of the Mad God, Leap Day, Steambirds). Michelle Clough, Ted Castronova, and Andrew Friedman have decades of experience on related design topics including MMOs, economics, education, and narrative. After Project Horseshoe, we also interviewed Michael Molinari, a designer at the bleeding edge of games played over Twitch (Choice Chamber).

What is a Shared Avatar game?

Shared avatar design involved **multiple players** controlling or influencing a **single avatar** such as a human, a monster or a starship. Together they must **collaborate** and **negotiate** to **accomplish** a **shared**

goal. Examples include Twitch Plays Pokemon or Artemis.

The player's mental test is to look at the entity on the screen and ask if the player would identify it as, "**That is us**" This paper focuses primarily on Shared Avatar games.

What is a Crowd game?

Crowds happen when **multiple players** have individual agency - a vote or a single move or point of leverage - but the combination of their moves looks like a **larger emergent entity**, like a crowd. A crowd game gives individual players tiny things to do but their actions create a larger impact. That larger phenomenon is not created by the designer (to be filled by player actions, as in the shared avatar example) but emerges directly from gameplay. Examples include Realm of the Mad God where a mob of players acts as a self healing super protagonist.

The player's mental test is look at the entity on the screen and ask if the player would identify it as "**That is me.**" This paper briefly touches upon Crowd games.

Why design Shared Avatar games?

Shared avatar open up new opportunities, both financial and intellectual.

There's a lot of blue ocean here!

While there have been a handful popular innovations that involve shared avatars, there are very few dominant games. It is far away from AAA game development, mobile games and other highly competitive red ocean markets. For small teams interested in carving out their own area, this is one of the more productive and untapped areas of game design worth exploring.

Strong shared goals yield high retention, high engagement groups

By putting players together with a strong shared goal and a shared body, they are encouraged to bond together.

This results in highly social communal groups that engage strongly with the game and theoretically retain for longer periods of time.

The chance to design a society

When we build shared avatar games, we inevitable find ourselves designing game governments. Technology allows a designer to create an experience for an individual citizen, that aggregates into a collective decision about the actions of a larger organization. Designers must design the sorts of things that constitution-writers have to design:

- Who votes? What does the vote mean?
- Who decides what to do next?
- How do we find out what other people want?
- How do we aggregate those desires?
- How do we solve conflict?
- Who gets to communicate with others, and how?

We are making games that explore the systems and tools of politics. Greece invented democracy. What rich political systems will game designers invent?

The chance to design a higher shared purpose

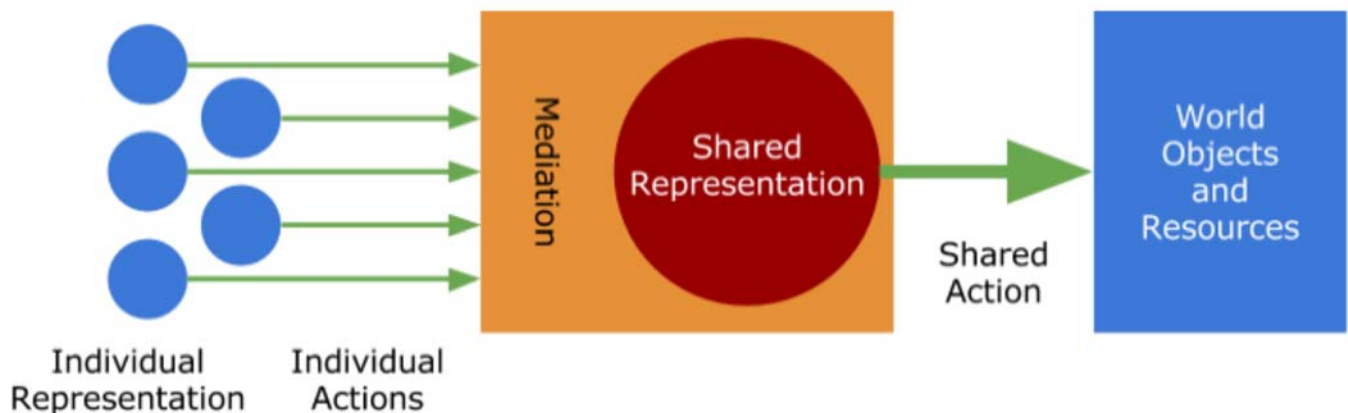
The same questions apply in the space of meaning. A designer can create an experience and shared purpose bigger than the individual for a player, then aggregate that experience in a way that any outsider would recognize as a religion. In doing so, the designer draws lines that, in other contexts, have brought souls closer to gods, and vice versa.

Feeling oneself to be part of a larger whole is perhaps the most common mystical experience. In this design space, the designer touches on powerful questions about the basic nature of human reality and meaning. On first blush that's merely a fanciful head trip, but

Let players experience different scales of mental and social hierarchies

People tend to see themselves as a self contained individual within position in the hierarchy of society. But they also sense the Inside Out collective nature of their consciousness, and their membership in larger self contained entities like families or nations. Moving people up and down these hierarchies is a way to help them think critically about mental and social hierarchies and organization scale.

Key elements



There are several conceptual elements of Shared Avatar Games. By naming these, we hope to build a shared language around the top.

1. Shared Avatar
2. Shared Goals
3. Mediation of Collective Decision Making
4. Emergent Player Organization

1. Shared Avatar

The first requisite characteristic of a shared avatar (SA) game is the presence of the common avatar. This typically requires:

- **Shared Representation:** A physical/visual representation in the group; a shared body.
- **Shared Actions:** A set of functionality or capabilities that the avatar possesses and can perform (typically mediated through the individual player actions).
- **Individual representation:** Does each individual player show up in the game somehow? Often this is

as simple as a name in a shared chat stream.

- **Individual: Actions:** What can each individual player do?

Spatial representation

Players are inside the avatar: The more common format for SAs is a single, large object/structure/vehicle /creature that all of the players are contained within. Spaceship, mecha robots, naval capital ships or submarines are all classic examples. Control or direction of the avatar's actions is typically represented through the players' use of control panels, work stations, steering wheel/tiller, etc.

Players are outside the avatar: Another pattern is where the players are external to the avatar and control its actions indirectly. Individual actions could include physical activities such as pushing/pulling or verbally directing/instructing. Twitch Plays Pokemon involves external players giving orders to the shared avatar. Control could also be represented through impacting the game world in ways that an AI-controlled avatar then responds to: blocking movement paths or drawing its attention to objects, putting them in front of the avatar, having them making noise/flash/etc.

Individual vs Shared Avatar dichotomy

In most cases we considered, the player has a visual representation in the game apart from the SA. There's a sliding spectrum here between fully embodied individuals (as we see in crowd games) and abstract and anonymous individuals. Players typically crave representation and individual acknowledgement so even the briefest handle can be a strong motivation. This can lead to a state we referred to as "**low mental unity**" - where players are more concerned with their own goals and experience than that of the shared group.

This creates a design challenge around ensuring that players experiences the game primarily through the filter of an "us" experience rather than a "me" experience. For example, players may fall into individualist thinking in a game like Artemis where each player has a defined role and the ship is more of just a shared tool.

Something that comes up repeatedly from designers of crowd or shared avatar games is how important it is to highlight some individuals. Even in the context of a strong shared goal, players latch the following moments:

- **A means to express themselves.** Players want to say something and be heard. Even if it is just doing a simple cheer animation (1vs 100). Molinari calls this "Giving people a voice".
- **The opportunity to be in the spotlight.** Brief moments of celebrity are incredibly powerful. In Choice Chamber, Kickstarter supporters have special powers that only they can activate. This causes all the other players to fawn over them and want to be them.
- **A persistent identity:** Humans constantly engage in relationship-building reciprocation loops. In order to do this, they need to have a persistent handle. And the more singles of past behavior or accomplishment that can be built into that handle, the stronger the social proof that a relationship is worth pursuing. A persistent chat nickname with a badge for helping others is more engaging than anonymous voting.
- **Massed individual expression is still a powerful shared moment.** If crowd of rainbow colored people is moving of their own accord in the same direction, it can feel extra meaningful.

2. Shared Goals

The way to encourage "us" thinking - what we termed "**high mental unity**" in thought - is to make sure individuals have strong shared goals. These give players something to work towards.

For example:

- **Shared Survival:** If the shared avatar dies, we all die. In a bridge simulator like Artemis, when the ship is destroyed, it is assumed that all crew members also perish.
- **Shared Success:** If we work together, we will be more successful.
- **Shared Entertainment:** Intriguingly, games that go horribly wrong are often just as fun as those that go well. If the result of player actions is a catastrophe, or results in unexpected outcomes, the group as a whole will often try to make such moments happen again. In Realm of the Mad God, mobs of players known as 'trains' would often run headlong into enemies that were far more powerful than the individuals. Most would die.

A key source of shared goals is communal resources. Players can't satisfy all the individual needs due to limited actions, time, or resources (mana, health, etc). The shared avatar forces a level of communal resources that everyone must manage together. Attempting to direct those limited resources towards some end immediately generates shared goals for the group.

3. Mediation of Collective Decision Making

All shared avatar games need a conflict resolution system. It is rare that everyone will agree. It could be democratic - total the votes. It could be random, or based on pseudo-random factors like, the game selects the first command it gets after the lag settles down. Or the designer could create specific roles that parcel out asymmetric information or control. For example, the Engineer can stoke engines, but she is the only one who can do that. But the Engineer does not know where the ship is, only the Captain knows that.

We have divided methods for player's collective decision making processes in controlling the SA into

- **Indirect:** The individual player's actions are aggregated and filtered by some system before they are executed by the avatar. Most often some form of voting.
- **Direct:** Direct means that the player acts directly on the avatar or the avatar's environment (although often within a limit role-based scope or sphere of influence).

Indirect: Voting

Voting systems can encompass a wide range of formats and structures (and can but do not have to be a formal ballot process). This is an extraordinarily rich topic with a long history; too much to actually cover in this paper. We instead focused primarily on untapped techniques unique to digital Shared Avatar games.

- **Method of aggregation:** the method of aggregating user votes can be loosely categorized as representational or nonrepresentational.
 - **Representational** systems follow traditional democratic style voting methods of aggregating player votes over a period of time and then selecting a choice (typically the most popular one), at which point the votes and action choices reset and voting can begin again. Variants might allow increased or decreased vote power over time based on the results of previous votes or some characteristic of a player (e.g., seniority).
 - **Non-representational** still make use of player "votes", but in more atypical ways that are not consistent with established voting system mechanics. For example, Twitch Plays anarchy mode, where votes (actions entered into the chat window) are entered into a simple FIFO queue, and acted upon in order until all are complete. Other forms of programmatic/algorithmic responses, might include picking a random player vote, or assigning voting weights/priorities based on non-standard factors (timing, geo-location, etc). Alternately, a designated player, GM, or moderator reviewing the voting choices can manually pick a vote choice.

- **Time period for aggregation:** If the voting method aggregates over time, then the period used can significantly impact the gameplay feel. 1 vs 100 represents the extreme short end of the time period spectrum where vote input is aggregated over less than one second. Twitch Plays “democracy” mode aggregates over a 10-20 seconds. A long-play, turn-based game could potentially aggregate votes over a period of 24 hours or more.
- **Voting as decision resources:** In game systems, “votes” can also be considered as a conceptual framework for use and transformation of limited resources (see [Machinations](#)). Instead of players simply voting for a action decision they support, actions can be determined by players assigning resources or energy they have access to, time spent performing sub-actions, etc. Computer mediated and arbitrated voting systems in games can be far more complex than an Athenian democracy, so take advantage of it! Useful examples
 - **Votes that return to the player** if the option they select isn’t picked. Lessens the sting of being the losing side.
 - **Votes that convert into other resources** based off losing. If you do badly at one area of the game, you gain opportunities in another area of the game.
 - **Vote accumulation:** Trigger actions when enough votes are accumulated, not at the end of some period of time. If players can contribute 200 votes to a particular project, it will occur.
 - **Vote decay:** Players can collect a number of votes, but if they don’t use them, they start to lose power.
 - **Vote differentiation:** Different votes are used for different actions. A player may have votes for an attack action, but they don’t have votes for a building action.
 - **Vote collection:** Instead of just giving a player a vote, they may need to collect it by performing some action in the game. They could harvest votes like one harvests lumber and then use those on crafting actions.

Common problems: Voting as a decision making system introduces some risks into gameplay. Due to the lag allowing players to review and evaluate options and then collect votes, pacing can become an issue. From a social dynamics perspective, a player consistently on the losing side of voting systems often become disenfranchised leading to disruptive behaviors and decreased player retention.

Direct: Role based

The role based, direct input structure is a fairly common paradigm for games in this space, as well as being easily recognized in other forms of media. The classic example of the bridge crew on the USS Enterprise in Star Trek. Each crew member sits at their own “station” with a distinct set of controls related to their role on the shared avatar of the ship.

There are many pre-existing, commonly recognized / stereotyped roles everyone automatically knows how to act. One well known example is the Zimbardo Stanford Prison Experiment (https://en.wikipedia.org/wiki/Stanford_prison_experiment), in which groups of college students volunteered for a psychology experiment where they were placed in the role of prisoner or prison guard. Participants rapidly adjusted their behavior to fit the role to such dramatic degrees that the experiment was ended early. Another example of role Tropes is Kyle McDonald’s Facial Substitution/Re-mapping. This resulted in the use of expressions that map to the behaviors commonly associated with the celebrity’s public persona. <https://vimeo.com/29348533>

Common “core” roles for games of this type are often modelled around traditional squad / submarine culture (see also Command Roster trope: <http://tvtropes.org/pmwiki/pmwiki.php/Main/CommandRoster>) and can include the following:

- **Captain:** Some role-based games will feature a Captain or Leader role, who is typically responsible for directing the actions of the other team members. The captain may have access to additional information required to form a holistic view of the SA, or may serve as a conduit of information between the different specialists. The captain may also have final decision making authority over the ordering of actions or allocation/prioritization of shared resources across the SA. Staying with the Star Trek example, Captains Kirk and Picard gave orders to specialists at the various stations and provided sequencing, but didn't perform any of the station actions directly. They may also handle external communications as the "voice" of the SA.
- **Specialist:** the specialist has responsibility for over a subset of the shared avatar's functionality. The distinct actions the specialist can perform allows them to control or affect a limited sphere of the shared avatar. This can be realized as interaction with a "control panel" or management of a type resources.
- **Support:** Unlike the specialist, the support role type does not have a work station or similar mechanic to directly control an SA subsystem. Instead, their direct abilities revolve around enabling or empowering specialists to perform their role. The support character may control complementary resources, provide buffing, etc.

In addition to these primary role types, there are a number of less common roles including:

- **Traitor:** the traitor role is working at cross-purposes with the other players (often covertly). This role type may have responsibilities/controls comparable to another character type, but with a different victory condition requiring them to use those controls in a different way. Traitor roles can also often be found in co-op board/card games (e.g. Resistance, Spyfall).
- **Spectator:** the spectator role falls outside of the previously used "crew member" analogy. The spectator has an outsider view of the action in progress (often an omniscient / god's eye view, but sometimes restricted like a specialist), but are visually represented within the SA. Spectator's ability to influence the SA, typically occurs in a secondary or tertiary manner, and often can only be done collaboratively. Dropping "resource packs" to Hunger games combatants could be seen as an example of this type of role. Spectator roles may also be entirely passive until they are "promoted" to being an active crew member (see hero moment).

Direct: Non-role based

Direct input games can also allow each player the full range of actions or avatar controls. For example in a driving game, each player could have his "hand on the wheel" and apply pressure to turn it in a certain direction (or prevent it from turning, apply pressure to brakes or gas, and so on. Games in this format, can also work with external avatars. In a (theoretical) "Earth Ball" game, each player has equal ability to push/pull/grab the ball participating in controlling its direction of movement.

	Single avatar / Inside the avatar	External avatar / Outside the avatar
Voting	Twitch Plays Pokémon	Hypothetical version of <i>Choice Chamber</i> with AI heros.
Roles / Direct Input	<i>Artemis: Spaceship Bridge Simulator</i>	Hypothetical multiplayer <i>Nintendogs</i> or <i>Black or White</i>

3. Emergent Player Organization

To successfully operate a shared avatar and reach a shared goal, a group of player must adopt a certain level of shared language, trust and social norms. This does not happen instantaneously and often needs to be cultivated.

Stages of Development For Groups

One standard model for this from organizational psychology is Tuckman's Stages of Development for groups. (https://en.wikipedia.org/wiki/Tuckman%27s_stages_of_group_development) He postulates that groups go through several stages:

- **Forming:** Group comes together and forms under an official identity. Often this happens when players are matchmade into a game room or scenario.
- **Storming:** The group has different expectations, language, social norms and schema for seeing the world. When they attempt to perform, they end up clashing. Politics, power struggles and other dramatic player interactions occur at this point. Often the group fails to perform at this point.
- **Norming:** Players start to agree upon shared language and polite behaviors. They understand individual and group boundaries and develop trust. Expectations are rewritten.
- **Performing:** Now that the group has the social tools and trust to work together, they are able to finally tackle shared goals successfully.
- **Mourning:** The group breaks apart and goes onto other activities. If your game cares about retention, this is the dangerous moment where you lose players and spectators.

One critical design activity is that a game should explicitly design for this group formation process. Just as we design an RPG leveling curve for how a player's in game power will increase, we can design a progression system players that encourages to reach a state of coherent group performance.

Tools for managing group formation

Here are some tools that existing in shared avatar games that help facilitate this

- **External Pressure:** The group has a strong external pressure (environmental or another group) that is causing them to lose resources or get further away from their goals. Only by self organizing can they push back against this external pressure.
- **Ramping Friction:** As the group gets more powerful when they work together, the system pushes back even stronger. This keeps the group performing.
- **Non-zero sum systems:** Players do better when working together.
- **Safe group actions:** A new player can do something that is part of the group that is a simple, low execution skill activity. In general, you want new groups to perform low execution skill verbs and mature groups to perform high execution skill activities.
- **Soft coop vs Hard coop:** Soft coop means that players who work together are not blocked if others don't help out. In Realm of the Mad God, if one player attacks another monster, it is nice if someone helps, but not necessary. Hard coop means that there is a hard dependency on another player. In many coop Zelda games, one player must step on a switch for other player to pass that section. Also new groups perform low dependency activities and mature groups perform high dependency activities.
- **Hazing:** Unintuitively, creating an environment that pushed new players through some form of ritualistic distress where they must reaffirm their membership with the group and support of the group's shared goals tends to increase group coherence. In cross cultural studies the stronger the hazing ritual, the greater the group retention.
- **Censure, Oothering and Shunning:** Groups that have members that are not performing or are actively working against the group will tell them how to improve or kick them out. Give them tools to do this that are fair to both the group and those that are targets of their issues. This system is both an opportunity for wonderful gameplay (traitors) and an often overlooked area that results in griefing.

Other factors that impact group formation

There are a couple other factors worth considering when determining how much work you'll have to put into

the onboarding process.

- **High bandwidth vs Low Bandwidth:** Artemis is often played in a living room with live people. Board games involve people sitting face to face at a table. These are high bandwidth social interaction scenarios where players rapidly communicate using speech, tone, facial expressions and body language. They tend to very quickly reach shared group norms. This class of game often doesn't even bother with the onboarding process because it happens so naturally. The flip side is low bandwidth scenarios like you find with online games. Here players are communicating through either in-game actions (like Journey) or simple text. Groups convert much more slowly and each step of the process must be carefully managed. A classic mistake is to test a game in a high bandwidth environment and then deploy in a low bandwidth environment and wonder why no one is having fun.
- **Stranger vs Friend:** Some groups come in with pre-existing norms. These are friends or working groups. Other groups are a mix of unattached strangers. The strangers need more hand holding in order to create a performing group.
- **Worst of all worlds:** Often you create groups out of strangers online. Due to lack of trust and low bandwidth negotiation channels, they become stuck in the storming phase and never make it into the performing phase. This is the group that most needs a group formation progression.
- **Amplification and simplification:** When groups are trying to enforce norms in a low bandwidth environment with high execution requirements, non performing members risk censure by the group. However, there are few positive tools so they fall back on degenerate techniques. Amplification is saying the same thing repeatedly, or louder, or with more emotional emphasis (swearing). Simplification is reducing the complexity of the norming lesson so that it can be easily digestible. Both these explain the behavior in League of Legends players when a player doesn't respond. In a high bandwidth environment, we tend to see more nuanced reciprocative methods with listening and give and take that help teach norms in a less blunt fashion. This is also an opportunity for adding gameplay since designers can create positive norm enforcing signals that are both amplified and simplified without falling back on aggressive swearing.
- **Group Size:** We interviewed Michael Molinari, creator of Choice Chamber and a key learning was that the same game varied dramatically based off team size. He categorized teams into small (1-10), medium (11-50) and large (50+). His game has been played by teams of up to 45,000 players at once. In smaller teams, much more collaboration occurs. One team of friends was able to speed run the game in 5.5 minutes vs the typical hour long playthroughs. Impressive, but if norm formation is where the fun of the game comes from, this may not be all that entertaining. On the other side are the large games. These give this highly communal feeling of being part of a vast crowd, but have issues with identity loss. No one cares if you leave the game if you are 1 of 10,000. No one cares if you contribute or not. His preference is medium sized teams that have a good mix of drama and chaos, but this is ultimately a personal player choice.

Emotional Outcomes

What are the emotions that shared avatar games produce? Unlike many single player games, shared avatar games target a slightly different set of key motivational factors from [self-determination theory](#):

- **Mastery:** Shared avatar games involve a strong activation of social skills, an area uncommonly developed in more traditional digital games. Players develop critical skills in cooperating, forming shared language and norms.
- **Connectedness:** Shared goals help players connect to a higher shared purpose.
- **Autonomy:** Shared avatar games often weakest at expressing autonomy. Large group voting systems or hard cooperative games are particularly problematic. Crowd games and role-based shared avatar games are an interesting compromise that give players strong autonomy in the face of overwhelming shared goals.

When people participate in shared experiences, there are various behaviors that emerge.

Trolling

Trolling is intentionally working contrary the group's goals, either to sabotage efforts or draw attention to themselves. This stems from the desire for Autonomy: individual power and rebellion against the group.

Celebrity

Also in games, players will also have the desire for celebrity, the chance to be a star, a hero or a director. A star wants to show off skill or talent, a hero wants to be the sole actor in a key step in a story like conquering the dragon. These tap into the desire to share mastery. The director wants everyone to listen to their orders. This come from the desire for Connectedness, although desire for some Autonomy (as "the hero") also informs it.

Mitigation

Trolling and Celebrity are often closely tied. They are both about the self in the face of a group. There are a few ways to mitigate their impact on a group. Trolling can be stopped aggressively with limiting communication or policing behavior with banning. It can be dealt with passively through giving outside channels for players to comment and heckle, or player controlled muting of others. Celebrity is also a way to mitigate trolling. By offering the potential for a player to become a famous for a short time, many players will hold off their misbehavior as to not interfere with their chances or save their comments for when they have their spotlight.

Individual accomplishment can be public facing or not, but it gives different emotional rewards than group accomplishment. Individual hero moments can be a firm measurement of Mastery, whereas with a group, success or failure is harder to assign to individuals. Group success is very strong emotion. This deep satisfaction that is a foundation for human society. Acting as a group, at its best, can satisfy Mastery of skill, Autonomy through roles, and Connectedness through teamwork.

Problems

There are a variety of common problems that arise within Shared Avatar games.

Lack of examples

These are a new form of games. If you come from a background of incremental improvement ("Like game X but 10% different"), there are no examples to borrow from. Molinari notes it is hard to design for this type of game because no one has done it. You spend a lot of time solving new problems and a lot of time just understanding how to formulate the problem.

Potential issue: Unpredictable preproduction and production schedules.

Lack of agency (voting)

Players can feel a sense that their vote, or their presence, does not matter in the long run and that they are doing little to directly affect or control their avatar. This is particularly problematic with players who vote for the "losing" side, even more so if they consistently end up on the "losing" side.

Potential issue: Sense of frustration, lack of agency, lack of connection to events and the avatar

Slower pace due to decision mechanics (voting+)

In a single-player game, the avatar can take actions instantly based on the input of the player. In shared avatar games, however, the action must be delayed until all players' inputs are collected and collated (e.g. taking a vote and tallying it). This can result in a slower paced game with potential pacing issues (stopping/starting), as well as delays without gameplay as players wait for the decision mechanics to resolve

Potential issue: inconsistent pacing, lack of forward momentum, frustration/boredom from players

Trolling behavior (direct input)

- In games where all players have direct input or control, it is possible for one (or more) player(s) to intentionally derail the progress to a much greater degree than voting - for example, an Artemis officer deliberately passing on misinformation vs a single Twitch Plays Pokemon troll
- As discussed, trolling is motivated by a need for rebellion against the group, for individual power, and to a certain extent, for celebrity - whether to be "the jerk" or to make people laugh

Potential issue: good feelings for troll, but frustration and failure for other players

Norm enforcement: Group shaming someone who does badly (direct input)

- Similar to multi-avatar co-op games such as League of Legends, shared avatar games can fail if a member of the team underperforms or performs incorrectly... and like such co-op games, may suffer from group shaming of the underperforming player
- This is the flip side of the celebrity/hero design of direct input or role based play; while an individual can be a "hero," they can also be a "failure" and be singled out as such.

Potential issue: Frustration among the "shaming" players, alienation of the underperforming player, possible source of toxicity in group and community dynamics

Concurrency and Logistics

- Playing together requires organizing player schedules. Online games require servers, lobbies, the right mass of player base currently logged in. And for local multiplayer, people's schedules have to coincide, a location has to be chosen, multiple computers/consoles/screens/etc may need to be brought... and someone has to go out and buy all the chips and beer!
- There is also added challenge if the game is not a mere one-shot experience; if a game is meant to be played over multiple sessions, adds to the logistical complexity and difficulty of making regular play sessions, continuing with the same players, etc.
- Not a problem specific to shared avatar games, but with the larger blanket of multiplayer in general

Potential result: Running a shared avatar experience or game is logistically challenging, particularly as the design may require a base number of players to even function. Imagine a Voltron game... "Can't play tonight, everyone, our left leg player had to stay home!"

Dynamic team sizes

Molinari notes if you have a large potential audience like many Twitch Plays games, you need to design for anywhere from 1 player to 100,000 players. The dynamics shift at each team size and something as simple as chat goes from being useless (team size of 1) to the heart of the experience (team size 2 to 150) to confusing (team size 150+ with a hyperactive stream of comments.)

Potential issue: Bad player experiences if you don't hit the optimal team size.

Complexity of building online games

The majority of shared avatar games are likely to be online, resulting in the same problems as online games in general - server strain and back end costs, latency, bandwidth, moderation, etc - all of which add costs to the project

Potential issue: Higher costs, Schedule overruns.

Freeriders

- Some players may refuse to participate in the game, neither helping nor hindering but just sitting back and doing nothing.
- Depending on the design of the shared avatar and the control method, this could be anywhere from annoying but non-interfering to causing failure of the group and the avatar; it's one thing if the freeloading player is playing the left eyelash, quite another if they're the right leg!

Potential result: Frustration from participating players, disconnect of the freeriding player (as they become more spectator than player)

Massively simultaneous input

In games with large spectator-style populations, you often have a lot of people doing the same thing at the same exact time. Servers typically aren't built for this since naturally a concurrent population will spread their actions out over time. A game like 1 vs 100 on Xbox Live was a massively concurrent game show where a timer would tick down and then everyone would enter their response in very narrow time window.

Either architect your hardware to deal with situation, reduce your group sizes so that standard hardware can deal with the load, or don't design games that require massive simultaneous actions.

Twitch has massive lag

With the success of Twitch Plays Pokemon and the promise of massively streamer audiences, Twitch or other streaming services are commonly seen as a critical platform for building shared avatar-style games. In practice, Twitch has a 15+ second lag between what happens and what is seen on the screen. This eradicates huge swaths of potential game designs.

The alternative is to mediate the entire experience through a traditional server - client architecture. This allows for more pacing, real time mechanics.

Potential issue: Many promising designs are not possible on Twitch.

Twitch's streamer audience is small

There are approximately 1 million Twitch streamers. If you are selling directly to the streamers, this is a much smaller audience than the 10s or 100s of millions buyers on other platforms.

Potential issue: If selling to only streamers, expect your overall sales to be low.

Getting viewers to pay the developer is difficult on Twitch

On Twitch, viewers only really have an option of paying the streamer. Even if millions of viewers are

watching your game, there's a good chance all the money will funnel to the streamers through streamer-specific subscriptions and donations, and nothing will go to the developers. For games that are not directly dependent on Twitch, this has not historically been a major issue since a viewer can convert to a player by purchasing a stand alone copy of the game. With Twitch only games, the streamers don't have anything from the developer to advertise or upsell.

Potential issue: You may have lots of viewers, but make no money due to current streamer-centric payment structures.

Live iteration on the design

Because many of these games are highly political and rely on community formation, they are designed much like any rich multiplayer game. You organize live playtests, you have an ongoing beta testing group, you update on a regular basis both before and after launch.

In highly emergent gameplay with mob-play, you'll often notice new behavior that you want to build the game play around. A process of community observation and amplification often works well. In Transformice, a mob-based MMO about a flock of mice reaching cheese (and failing hilariously), much of the joy comes from the odd troll-y behavior of the mob. They amplified this with special troll characters that then amplify the mayhem.

Future questions

Shared Avatar games are a wonderfully unexplored and sparsely populated design space. Though we have enough examples to know that something interesting is going on here, there are still numerous questions. The good news is that smart teams that find out answers will grab a new product market with very little competition.

- **What is the business opportunity?** How do we turn Twitch Plays Pokemon from a fad into something that makes money? Infiniminer can become Minecraft... what is the breakout hit for shared Avatar games?
- **Who wants these games?** What player audiences would enjoy these games?
- **How do we talk about these games to our potential players?** What is the hook that sells this style of experience relative to existing activities.
- **Cooperation or competition?** Do we leverage this to encourage unity, harmony, and cooperation among people (within game and in larger society), or do we also explore competition and conflict?
- **What are the ethical demands on people designing for shared avatars?** Is it important to give people a voice? Do we need to allow people to vanish into the mass if they wish?

Appendix A: Shared Avatar Examples

Twitch plays Pokémon

- Twitch audience took control of the PC in a game of Pokemon Red by inputting keywords for directions and interactions into Twitch chat; the avatar moved based on either an aggregate or a random selection of the commands
- high mechanical unity (one avatar), mixed mental unity - a midpoint of unity (the goal of beating the game) and individuality/dissent (trolling, anarchic chaos)
- utilized voting systems, both democratic and anarchic
- Dominant theme/takeaway: novelty, anarchy, ridiculousness

Artemis: Spaceship Bridge Simulator

- Players assume various roles on the bridge of a starship and work together to complete a mission involving navigation, combat, and exploration; each player's computer gives them role-specific information, tasks, and controls
- high mechanical unity (one avatar, the ship), high mental unity (all trying to complete the mission, sharing info)
- Role-based direct control - captain, engineer, weapons officer, navigator, etc
- Dominant theme/takeaway: living the Star Trek dream, working together to complete the mission/save the ship

Spaceteam

- Players use their phones or tablets to access different comedic consoles on a spaceship and receive a unique set of (ridiculous) instructions. Players must communicate these instructions to the correct teammates (e.g. one player may have a coffee-shaking dial, while another gets the message, "intensify coffee-shaking to five!" and must communicate this)
- mid-high mechanical unity (one avatar, the ship, though not seen), high mental unity (sharing information/tasks to survive)
- Role-based direct control - players have certain consoles on their device
- Dominant theme/takeaway - comedic, chaotic control of an exploding ship

Lovers in a Dangerous Spacetime

- Two players have to constantly shift roles to maintain a starship's systems. Piloting, defending, attacking, etc. Their ship is a small platforming level they must traverse between stations/roles.
- mid-low mechanical unity (one ship, but also avatars within the ship), high mental unity (falling into flow state and shifting roles together)
- Role-based direct control - each player takes on different functions in the ship
- Takeaway - high coordination and task swapping can produce a feeling of flow usually only felt in single player games.

Everyone is John (example of competitive shared avatar)

<http://overlycommonname.github.io/john.html>

- Players take roles of different Voices in John's head that want to push him into different Obsessions (e.g. dancing the can-can); to that end, they must vie for control.
- high mechanical unity (one avatar, John), low mental unity (everyone competing against each other for different goals)
- Anarchic voting control - game systems involve spending Willpower points and rolling dice to determine who is "in control"
- Dominant theme/takeaway - voices warring for control of a man

Choice Chamber

This is more of a shared environment game.

- Platforming game where the events, powerups and monsters and controller or influenced by ongoing voting by the audience via twitch stream.
- Low mechanical unity (one player to one avatar, the other players as spectators), middling mental unity (other players can help or harm the controlling player)
- Democratic voting control - occasional polls to see what happens next

- Takeaways - the player is often talking to the audience asking them to help him get specific items.

Upsilon Circuit

- This is a Diablo style dungeon survival game with permadeath. But it is run like a gameshow, only on for an hour a day, only one server. Players never get to play again once they die.
- Low mechanical unity (one player with one avatar; other players are spectators), high mental unity (audience united in assisting their contestant)
- Unclear control scheme, likely voting
- Takeaways - Intensity coming from scarcity of only getting one shot and the audience watching the game.

Heroes of the Storm: Cho-Gal

http://kotaku.com/crazy-new-heroes-of-the-storm-character-is-controlled-b-1741046668?trending_test_e&utm_expid=66866090-62.H_y_0o51QhmMY_tue7bevQ.5

- High mechanical unity - A single hero controlled by two players.
- Separate skill tree upgrade paths

Collaborative storytelling

- “You Awaken In Razor Hill” - collaborative text adventure on WoW forums
<http://thelittlestmurloc.tumblr.com/Chapters>
- Mearls: collaborative D&D adventure <http://blogofholding.com/mearls/mearls.php?action=about&m=boh>
- High mechanical unity (one character), middling-to-low mental unity (unclear goal beyond “what happens next?” with players engaging differently)
- Moderated control - players suggest actions, the author chooses and writes accordingly
- Dominant theme/takeaway - a narrative heavy exploratory experience centered on one character

Media inspiration

- Voltron
- Power Rangers
- Inside Out

Appendix B: Crowd Examples

Renga

This is a large scale cooperative game usually played in a movie theater. Up to 100 people are given laser pointers to point at the screen. The players are taken on a 90 minute space adventure where they must self organise into groups of 4 to 10 to activate different functions of the ship. Each function has empty spots on it and will only activate if lasers are pointed to fill all the spots. This produces a ad-hoc organization based on unfilled tasks similar to how bee and ants work.

Takeaways: great game for mental unity and separate avatars with limited trolling potential.

Realm of the Mad God

This is a massively multiplayer cooperative MMO dual stick shooter. You wander around a large world with

your avatar fighting various monsters. However, most of the time, a group of players hang out together and fight monsters together. You get these large mobs of 40 to 80 players rampaging through the countryside healing one another, soaking up damage communally and taking on large bosses that are far too difficult for any single player. Several useful lessons

- **Shared XP:** Non-zero Sum XP so if one person kills an enemy, everyone on the same screen gets the XP. This reduces the tendency for players to compete with one another. More players is always better.
- **Shared Loot:** If someone gets loot, everyone on the screen gets loot as well.
- **Trains:** A large mob of players that goes through the map together. No one player decides the direction. And the group has a greater health and survivability than any one player.

Appendix C: Future looking examples

That Kid's a Space Case

- A five-player Artemis-style game about tiny aliens running a kid-sized robot around middle school
- Each player operates a part of the "child" - the eyes, the ears, the brain, the speech, and the limbs
- The eye player receives visual images of what the child sees, while the ear player receives audio of what they hear; they must pass this information on (verbally or by text) to the brain
- The brain must give instructions to the speech and limb players as to what to say or how to move in response to this stimulus
- The goal: get through a day of middle school without making anyone suspicious

Voltron: The Game

- A five-player space combat game where players control the different mechs that combine to form Voltron
- The torso and each limb is operated by a separate player; they must coordinate their efforts in order to move, fight, etc

Choice Chamber... Minus the Streamer!

- A variant on Choice Chamber where the platforming avatar is completely AI controlled and will proceed through the level on autopilot - players act on the avatar by participating in votes on monster types and proliferation, level type and difficulty, powerups, etc.

All In: The Job Interview

- A two-team text game where players suggest and vote for different questions and responses within a job interview
- One team "controls" the interviewer, the other the interviewee. On one person's turn, their team must first suggest up to three possible questions/statements (first come first served), then the team votes on which one to say. The other team now must create and vote on their avatar's response, and so on.

Appendix E: Competitive Shared Avatar Design

Shared avatars lend themselves well to cooperative game design and working towards high Unity, but there is also room for competition and conflict-focused gameplay.

- Such a model would incorporate a single avatar “controlled” by multiple players with *different win conditions*.
- These games have high mechanical Unity, very low mental Unity, with a clear “winner”
- A current non-digital example is Everyone is John, an RPG where each player takes the role of a different voice in John’s head and attempt to direct him towards that voice’s goal.
- A key problem: how to resolve these differing goals/inputs? If control is handled by voting, everyone will constantly vote for their own outcomes, and a deadlock occurs; if control is handled directly / by role, runs the risk of all players sabotaging or stalemating each other and causing play to grind to a halt
 - Answer: Have the different win conditions share certain goals along the way, encouraging players to temporarily cooperate at varying stages

Illustrative example

A four-player game about a car where each player has a different destination in mind (a movie theatre, a restaurant, a gas station, a supermarket). Car steering/route driving is handled by player voting

- At one intersection, the theatre player wants to turn left because that’s the route to the theatre; however, it’s also part of the route to the restaurant, so that player will also want to turn left, creating a majority vote to turn left. Gameplay continues smoothly... though those two players will likely conflict at a later street corner!