The Framework of a Game Design (MDA framework)

Article · April 2022

CITATIONS
0

READS
733

1 author:

Tan Guo Xin
Technical University of Malaysia Malacca
1 PUBLICATION **0** CITATIONS

SEE PROFILE

The Framework of a Game Design (MDA framework)

By Tan Guo Xin

Abstract

In this article, I introduced the MDA framework. In game design, the Mechanics-Dynamics-Aesthetics (MDA) framework is a tool for analyzing games. It formalizes the consumption of games into three parts: mechanics, dynamics, and aesthetics [1].An approach that attempts to bridge the gap between game design and development, game criticism, and technical game research. I believe this approach will clarify and enhance the iterative process for developers, academics, and researchers alike, making it easier for all parties to decompose, study, and design various game designs and game artifacts.

1.0 Introduction

Game designers tend to reject a methodological or structured way of developing a game because it is widely believed within the field that one cannot survive without creativity. However, there are many specific methods/frameworks that can help with designing games. However, most of them are more oriented towards the analysis of the game rather than the design process.

Ontologies can help by defining properties, concepts, and categories that represent play areas. It will improve and mature the growing digital games industry by improving understanding of the field and supporting a structured approach to game design.

Currently, there are no structured ontology games that are widely accepted by the industry, nor are they used in academic settings to aid in the design and development of games. This is not only because the field is a relatively new area of work, but also because some aspects are difficult to create to support ontologies. The lack of ontology in the field reduces the efficiency of game research, and this inefficiency is magnified when designing games.

Game design and development are difficult, and the difficulty increases year by year, the industry continues to evolve and gaming technology becomes more complex. To continue making games that meet quality standards, companies need to improve the efficiency and effectiveness of the development process. Over the past few years, several guides, methodologies, and theories have been created to help analyze, design, or document games. Some as design tools, some as documentation tools, and some as for game analysis, but most fail in some way. Sometimes, they even contradict themselves when describing basic concepts. In the field of game design; for example, the game mechanics that are considered integral to building digital games do not have a single, clear definition.

According to Hunicke et al. [2], games are subdivided into different components: rules, systems, and fun, which are related to their design counterparts, mechanics, dynamics, and aesthetics,

respectively. Therefore, mechanics are considered to be the cornerstones of games because they are closely related to the rules of the game.

Therefore, the clarification of the MDA framework will help clarify the relationship between all abstraction layers and the emotional responses that can be invoked in the player.

2.0 Diagram of the MDA framework

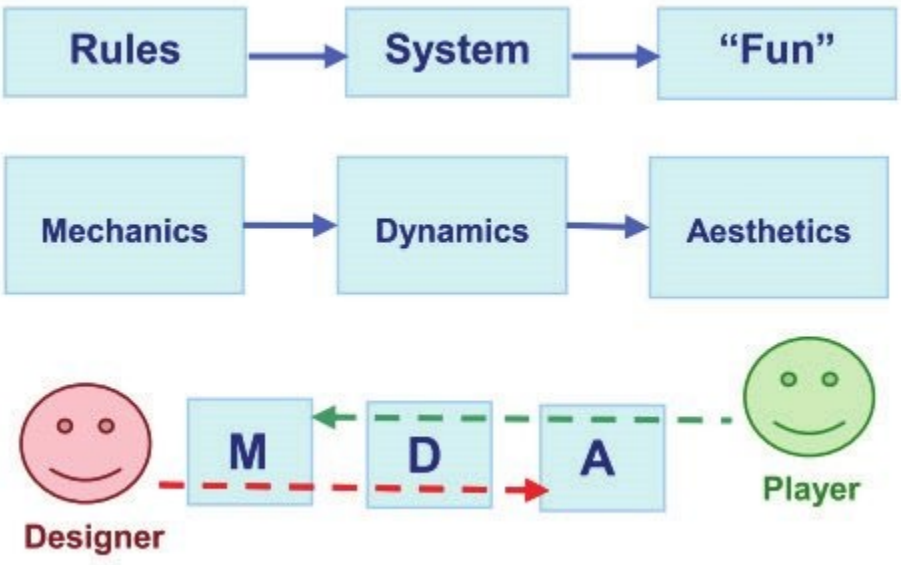

Figure 1: MDA framework order of influence

The closest to a widely accepted ontology is the framework proposed by Hunicke et al. [2], the MDA. It is influential and frequently used in universities all over the world. The MDA framework divides the game into three elements and proposes an order of influence between them (see Figure 1):

1. Mechanics: Describe the specific components of the game at the data level representations and algorithms.

2. Dynamics: Describes the runtime behavior of mechanisms acting on player input and each other's outputs over time.

3. Aesthetic: Describe what the player is like interact with the game system.

3.0 Description of each element in the framework

While somewhat accepted in the field, MDA is not used in the game industry, primarily for university analysis to aid game design efforts. According to the literature, the main problem surrounding this framework is the lack of scrutiny and accuracy of its concepts, and may even be self-contradictory in the definition. The framework has been criticized for going beyond its definition in certain aspects such as narrative, graphics, game sound, and interface, which may affect the invocation aesthetics [3]. Therefore, we recommend expanding it and clarifying a few things to make it more useful to game designers.

3.1 Mechanics

There are many definitions of mechanics in game design. For example, Sicart [4] proposes to define game mechanics as a way of expressing game structures and systems, with a vocabulary that allows for a formal and precise description of games. Thus, he defines "game mechanics as the world where agents call methods to interact with the game". This definition is based on object-oriented programming (OOP), where the mechanism of rules can be designed as methods and properties, and the agents are actors. The lack of conceptual precision points to a definitional problem: it is not clear what game mechanics are, supporting neither the definition nor the industry's acceptance of the game domain ontology.

The first part of the definition avoids the erroneous definition of solids as mechanics. In the previous FPS example, players, guns, enemies, and all described entities would not be defined as mechanics. If we take the famous game "Super Mario World" [5] as an example, the player is an entity that can jump - an action, or the responsibility of doing something - and knows its current speed of movement, or whether it is in the World Current Position - Know the Responsibilities. The responsibility to know is abandoned, not defined as a mechanism, and the responsibility to do requires one more step before it can be defined as a mechanism: examining its purpose.

The second part of this definition makes it clear that it must have one purpose: to invoke dynamics. This is important to avoid any unnecessary manipulations of game entities being treated as mechanics, such as game cameras controlling aspect ratios or UI responses to pointer clicks. These responsibilities have no direct purpose to invoke any kind of dynamics - they are just necessary to allow the user to interact with the game.

The importance of defining mechanics is obvious: designers can invoke direct control to achieve desired emotional goals or aesthetic dynamics. Identifying them is at the heart of game design and shouldn't be overlooked, nor should it take unnecessary time. This article wishes to propose a definition that makes it clear what should be defined as mechanics during the design process, and by doing so avoids unnecessary complexity and/or a large number of mechanics to define when developing a game.

To further strengthen the steps to identify and define mechanisms, it is recommended to subdivide mechanisms into three categories: implicit mechanisms, core mechanisms, and additional

mechanisms. The first representative takes on duties typically included in game genres, such as running, jumping, and dying on a 2D platform. If all these actions are defined as mechanics, the purpose of avoiding a large number of definitions of mechanics is defeated. We have to deal with these types simply: if it can be explicitly adjusted to call dynamics then should be defined as mechanics.

A core mechanic is a major action or responsibility in a game genre that is usually deferred to the main entity, such as a player shooting in an FPS or attacking an enemy in a fantasy role-playing game (RPG). These mechanisms are implicit and the most important in call dynamics, so they should always be well-defined.

The last category is extra mechanics. These are mechanisms that are usually defined later, sometimes after prototyping. This is the class of mechanics that we might call a game "extra", or the class of mechanics that would make a difference between similar types of games. An example is a camera shake or blurs effect in a horror game: this is an additional mechanism that can be defined because it has the explicit purpose of invoking one or more dynamics contained in the game. Mechanics are the only things a designer has full control over when trying to make a game achieve its emotional purpose or aesthetic by creating dynamics.

3.2 Dynamics

The MDA framework proposes a new dynamic definition to pursue a structured ontology of domains that can be used not only from an analytical perspective but also translated into the real game design world. The lack of a precise and unambiguous taxonomy does not support this goal. If the development team doesn't explicitly handle the dynamics that call them, how do they achieve the aesthetics of the game? There are two common approaches in the industry: relying on previous game-like dynamics and prototyping it in ad hoc ways until somehow the desired aesthetic emerges from it.

The first way is more common, and the consequence is impaired creativity in game design: teams are often unclear about how dynamics work together to invoke a desired emotional purpose (aesthetics) and to avoid risk, they rely on previous similarities. Replicate the dynamic technique in the game, especially if the team doesn't fully understand how it came about, or how it works on invoking aesthetics. This is not a healthy process for domains as it can lead to obvious similarities between different titles. How many RPGs are not dynamics of hunting monsters for in-game currency rewards (e.g., gold coins)? Such dynamics are so present in the genre that it's hard to imagine a game without them, but they do exist. A Game Boy game called Pokémon Yellow [6] was a huge success and created another incentive for players to hunt monsters: they could capture them, and the monsters the player captured could level up by gaining experience.

The second way is an inefficient way of dealing with game dynamics: while some useful dynamics only arise during prototyping and playtesting, without a clear understanding of how the mechanics create them, it will be nearly impossible for the team to invoke all the dynamics can be created to enhance the desired aesthetic, and it will take more time. If the taxonomy is well defined and understood, development teams can consciously study-specific mechanics to create the desired

dynamics, rather than relying on "luck" and hoping it will show up in playtests. Furthermore, of course, there is a greater risk of spreading unwanted dynamics in a released game.

The importance of dynamic coherent definitions is obvious, but not yet implemented by the game domain, especially when it comes to definitions that can be used in the design process. Due to its complexity and nearly limitless outcomes, there is no obvious benefit for teams to waste time trying to find them. This concept is mostly used for analyzing viewpoints: analyzing existing games and dynamics seems like an easier task to accomplish.

Dynamics is the bridge between the designer and the player. It's the action that emerges from the designer and creates an emotional response in the player. With a clear understanding of how mechanics underpin dynamics, and how dynamics create aesthetics, development teams can have a clearer path to achieving the emotional purpose of the game. A properly defined dynamically will support the design process by showing the development team where and how to work, increasing the efficiency of the development team.

3.3 Aesthetics

There is no clear way to determine what makes a game interesting. First, even defining fun is a difficult task: one can go deep into philosophy and psychology and still not find a clear definition. From psychology, we know that intrinsic and extrinsic motivation is what drives people to do things. Playing games for fun is intrinsic motivation. However, games have the ability to evoke intrinsic and extrinsic motivation. A good game must motivate players through extrinsic rewards.

According to Csikszentmihalyi [7], when a person performing an activity is immersed with focus, involvement, and enjoyment in the process of that activity, losing the sense of space and time, this person reaches the mental state of flow. Reaching the state of flow in a certain activity makes it an ’optimal experience’ since the user gets a high gratification from it. In order to keep a person in a state of flow, the activity needs to reach a balance between the challenges of the activity and the abilities of the user [8]. In video games, to keep the players interested (i.e., having fun), the game should be designed in order to maintain the players in the state of flow. Similarly, Lazzaro presented “The 4 Keys to Fun”, where she describes the main reasons why people play games: 1—Novelty; 2—Challenge; 3—Friendship; 4—Meaning. These four keys are related to the game mechanics, namely the hard fun, easy fun, the people factor, and altered states, respectively. Thus, it is crucial to have it in mind during the game design.

Based on Gnome Stew's website "8 Kinds of Fun" [9], these authors sidestep the difficulty of defining pleasure by introducing a taxonomy to rationalize pleasure and aesthetics:

1. Sensation: Play the role of sensual pleasure. A game with a strong sensory character - whether it's visual art style or sound design.

2. Fantasy: A fictional game. Games create a fictional world, a reality that players can choose from.

3. Narrative: Games are drama. A game with a well-written narrative, with a well-defined character or world.

4. Challenge: The game is an obstacle course. A competitively minded game that inspires the thrill of competition. It should be noted that it can also happen in a single-player game, where the fun comes from overcoming difficult challenges.

5. Friendship: Play as a social framework. One aspect of gaming is for players to build social relationships with friends, family, or other players.

6. Discovery: Gaming is uncharted territory. A game that inspires players to explore and discover new features.

7. Expression: as a game of self-discovery. Games that enable players to find ways to express themselves.

8. Submission: Gaming as a pastime. Games that focus on distracting players.

Based on this idea, the MDA framework proposes that the aesthetic definition should make it clear that players are ultimately responsible for creating their own emotions, and proposes the following taxonomy:

“Aesthetics describe the desirable emotional responses that the player can invoke when interacting with the game system.”

The first aspect to note in this definition is not directly related to the visual or artistic style of the game, but to the emotions that the player can evoke. The second point surrounding this definition is how it includes the runtime properties of the game domain it merges with: "when they interact with the game system". By including the runtime aspect, the aesthetics deal with the experience that occurs when the player interacts with the game, the MDA framework recommends making it clearer that the player is ultimately responsible for creating his own emotions: the game doesn't invoke emotions directly - it provides the tools and rules in the virtual world, Allows players to create their own emotions.

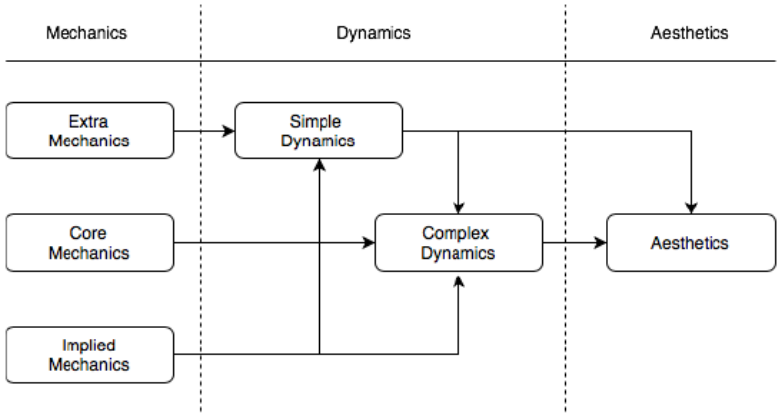

Figure 2: Diagram of MDA frame work

Understanding the aesthetics and how developers invoke them by properly following the suggested patterns (for example, Figure 2) will simplify the process of dealing with unexpected end-emotional outcomes: it's clear which mechanisms should be directly changed to improve the end result. Not only does it help with unintended consequences, but it also increases efficiency when creating new aesthetics, as how one layer will affect the clarity of the next layer - mechanisms (core, implicit, extra) create dynamics (simple) that call aesthetics , complex) . As a result, the relationship between mechanics, dynamics, and aesthetics is redefined, giving game designers greater control to design emotional responses that players can invoke.

4.0 Types of mechanics that can be used based on the elements in the framework

To demonstrate the analytical usefulness of MDA, we discuss the design process and illustrate some scenarios for commercial games.

The first step in developing a game is to define the core aesthetic of the game - or what the main experience the game allows the player to invoke. Sometimes this is a subjective decision: for example when the game idea comes from the inspiration or dream of a small indie development team. The main aesthetic here comes from the designer, and the definition can be backed up by analyzing similar games as a source of inspiration. In some cases, the aesthetics are predetermined by the stakeholders. For example, if a team is hired to make a sequel, an advertising game, or an educational serious game, the aesthetics can be determined by the contractor. The idea here is to define a primary aesthetic to support the development process.

When first defining aesthetic priorities, designers know where to put more effort into the design process. Hiring an entire orchestra to compose and record a soundtrack for a puzzle game as a pastime is not the best option (for most game companies). Submission is the target aesthetic here, not feel. Neither has assigned all the graphic design teams to create an extremely detailed environmental world in an online competitive racing game where the challenge is more important than fantasy and discovery. In other common scenarios, the team may decide to work more in a detailed 3D world, rather than redefine the mathematical progression of player attack statistics in an RPG, as the primary aesthetic may be fantasy or discovery rather than a challenge. The main aesthetic is friendship since it's a cooperative game. Followed by feeling (cartoon) and challenge (FPS online). To support this aesthetic, dynamics should be defined, and the mechanics that support them adjusted from there.

After the aesthetics are defined, dynamics should be proposed to achieve these emotional goals - and from these dynamics, the team can define and adjust the mechanisms that invoke them. Based on the proposed aesthetic and genre (FPS), teams can consider defining simple dynamics like "kill enemies" and complex dynamics like "defeat bosses as a team" and "group loot hunting". From there, it's time to investigate the mechanisms involved in these dynamics in a way that supports the intended aesthetic. Starting with the simple dynamic of "killing the enemy", the team can work on mechanisms that support this dynamic, such as "shooting" provided to enemy entities. Among these mechanics, teams can create different and more detailed animations so that enemies take

damage when shot by more than a specified number of players: it improves not only the fellowship aesthetics but also the feel. If for some reason (like not having enough animators, or using an engine that doesn't support that) this change is not feasible for the team, the team can create different audiovisual effects the moment the enemy is hit, because it also aesthetics can be improved. Another option is to make enemies more vulnerable from one side than the other, so the more players there are, the easier it is to kill, increasing friendships and challenging aesthetics.

Moving to complex dynamics, teams can handle the defined dynamics "defeat the boss as a team", which requires both mechanics (e.g. "player shoots") and dynamics ("kill enemies") in their creation. By always keeping the desired aesthetic in mind, this is done by studying the mechanisms that directly invoke these dynamics and the mechanisms that support it indirectly - in this case, the "kill the enemy" mechanism. Some ways to enhance it could be to adjust the power provided to the boss by the "being hit" mechanic that indirectly supports it, and make it take additional damage (Friendship and Challenges) when shot by two or more players in a short period of time. To improve the dynamic is direct engagement ("player shooting"), they can have the UI show all other party members how many times they have to shoot the boss and deal extra damage after the party member hits it, as it also increases friendship and challenge.

During playtesting, squad members would sometimes move away from their friends if the team noticed unexpected dynamics, detracting from the aesthetics of the friendship game. When an unexpected dynamic occurs, the team has some options: they can remove it from the game, ignore it, or keep it, depending on how it affects the proposed aesthetic. If an unexpected dynamic doesn't affect the desired emotional response, developers can choose to ignore it. An example is a game from Bethesda Studios called Elder Scrolls Skyrim. In this game, players control an avatar in an extremely detailed 3D world filled with creatures to defeat, caves and secrets to discover, and quests to complete. Exploration and fantasy are of course this game. No matter how many dynamic players a designer expects and creates, we must remember that the player is the ultimate creator of their own experience: the game is nothing but a tool. Players can create dynamics, such as trying to reach the highest point of the map, just to do that. This is (probably) not an intended dynamic, and it can of course be ignored as there is neither the potential to be redefined to support the desired aesthetics nor prevent players from implementing them.

To keep the gameplay as it is, removing the loot dynamics or combat dynamics might not be a good choice, but they should be better balanced to take full advantage of the game's detailed 3D world and narrative. Knowing MDA will help developers properly map the dynamics that cause this aesthetic imbalance and define what mechanisms they should change/remove or even create to address this. The dynamics of combat can be changed more easily so players need to rely less on items on the ground and absorb more of the narrative. Mapping entities can be reworked to separate the combat area from the narrative/immersion area. They can create new mechanics that provide narrative-related "points of interest", such as finding posters or NPCs and rewards for unlocking new dialogue, to motivate players to seek them out.

The purpose of MDA is not to create right or wrong rules in game design. It is intended to serve as a guide or blueprint to help developers understand how they can directly affect or change aspects of the game (mechanics), how they work together with player input over time (dynamics), and how

players interact with the game the emotional response (aesthetics) you get when you're doing it – and by doing so, it improves the final quality of the product.

5.0 Conclusion

Difficulty accepting structured design methods is a common attitude among designers in fields where creativity is at the heart of the creative process - such as music, film, literature, and games. This fact adds to the complex aspects of game design and makes it difficult to create a design approach. This paper attempts to propose a structured approach, supported by a clear ontology that does not harm creativity and actually helps it. By linking aesthetic goals to all layers of abstraction of the game under development, we hope to justify design decisions - as subtle as camera mechanics (e.g., shake and blur effects) in horror games, or as subtle as changing core mechanics As big as the purpose of the game - the anesthesia target, or the emotional response the player can evoke. By supporting designers as a guide during their creative process, they can properly target their creations to the intended outcome of the product.

A redefinition of the MDA conceptual framework has been proposed as a way to facilitate its use when designing games. Therefore, its main purpose is to eliminate the main problem of MDA and make it useful for game designers. Implementing a structured approach in a domain with so many specific features and aspects that must be supported by it is a daunting task. It's understandable that game designers won't easily adopt one. We hope this paper brings the field closer to this goal and allows designers to improve the quality of the development process and the number of players who have fun with it.

The authors of the MDA point out that the eight pleasures are a starting point for a vocabulary that can be used as a guide for understanding player emotions. By uncovering more of this subjective area of the gaming landscape, designers will gain a better understanding of the game's emotional goals and hopefully improve its quality. A paper by Roberto Dillon [10] enhances the way designers deal with player emotions. The author created what he calls the 6-11 Framework, a method that can be used with MDA, which focuses on six recurring emotions and eleven instincts in psychology: fear, anger, joy/happiness, Pride, sadness and excitement, and instincts are survival (fleeing), self-identification, gathering, greed, protection/care/nurturing, aggressiveness, revenge, competition, communication, exploration/curiosity, and appreciation of color. The additional detail of extended modeling of player emotions is a plausible way to further improve the effectiveness of game design methods, and future work is needed in this area.

Reference

[1] Wikipedia. MDA framework.
https://en.wikipedia.org/wiki/MDA_framework

[2] Robin Hunicke, Marc LeBlanc, Robert Zubek.
https://users.cs.northwestern.edu/~hunicke/MDA.pdf

[3] Design, Dynamics, Experience (DDE): An Advancement of the MDA Framework for Game Design
https://www.researchgate.net/publication/315854140_Design_Dynamics_Experience_DDE_An_Advancement_of_the_MDA_Framework_for_Game_Design

[4] Sicart, M. Defining Game Mechanics. Game Stud. 2008, 8, 1–14.
http://gamestudies.org/0802/articles/sicart?viewType=Print&viewClass=Print

[5] Super Mario World, by Nintendo. 1990.
https://www.nintendo.pt/Jogos/Super-Nintendo/Super-MarioWorld-752133.html

[6] Pokémon Yellow, by Game Freak. 1999.
https://www.nintendo.pt/Jogos/Game-Boy/Pokemon-Yellow-Version-Special-Pikachu-Edition-266142.html

[7] Csikszentmihalyi, M. Flow: The Psychology of Optimal Experience; Harper and Row: New York, NY, USA, 1990.
https://mktgsensei.com/AMAE/Consumer%20Behavior/flow_the_psychology_of_optimal_experience.pdf

[8] Nakamura, J.; Csikszentmihalyi, M. The Concept of Flow; Oxford University Press: Oxford, UK, 2002.'
https://nuovoeutile.it/wp-content/uploads/2015/12/2002-Flow.pdf

[9] Gnome Stew. 8 Kinds of Fun
https://gnomestew.com/the-eight-types-of-fun/

[10] Dillon, R. The 6-11 Framework: A New Methodology For Game Analysis And Design.
https://d1wqtxts1xzle7.cloudfront.net/14091846/GameOn-Asia_02-Dillon-with-cover-page-v2.pdf?Expires=1650296448&Signature=W-UOhT9IHyEUUa6rdMHJpl3m6hjCeaCzo48GzF1ByTijgup3aaIJLcEyna0JPeArtRwv2s1NIdJabVrWp8~M2YAPjmiKbRffHTekXOblwcxS8Lytd2m54agLnk1PxGsJE4HqCMp4F5F-3td3zRlB1brJTzCmLo7ghSFODn1YsSrcfVFE3fwk5LViBlkwEAmN2WUK1HWIwqlffYFQcVXXHVJFmgBiHZX3-Hr8DCtVRLQ1nWJHw6PXEstLKl-fpXLWjohMdxsDL4wmNwkjRyXCAjllgixaplwP~4wNHSefGZ8NHz2mQx36iArVUF4eSlPbfIqFUSDgU1oQD9N4JksZYA__&Key-Pair-Id=APKAJLOHF5GGSLRBV4ZA