

[gamasutra.com](https://www.gamasutra.com)

From MDA to DDE

27-34 minutes

The following blog post, unless otherwise noted, was written by a member of Gamasutra's community.

The thoughts and opinions expressed are those of the writer and not Gamasutra or its parent company.

Necessary enhancements on a commendable design framework

by Wolfgang Walk

After decades in game development, and after deciding to write a book about the role and rules of narrative design within the greater game design process, I recently found myself thinking intensively about the MDA framework. In doing so I realized there is currently no way to implement narrative design structures into MDA, and for that reason alone (there are others) I believe it's time to update the MDA framework. Not because it's useless, but because it is not the optimal tool it could be.

In 2004 I watched a GDC talk by Marc LeBlanc, who worked on Ultima Underworld 2, System Shock and Thief. The talk was about a formal approach to game design – and the noise you

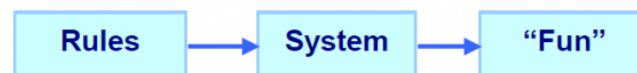
could hear during the talk came from dropping jaws hitting the floor. Here was someone who had delved deeply into the structures behind the design process. We all knew there must be a structure behind game creation – or at least we hoped there was -- and suddenly someone was showing us that structure.

Yet maybe the most important takeaway for me back then was: yes -- we could do it. We could understand what a game was on an intellectual level instead of finding our way by feel.

An accompanying paper, published the same year by Hunicke, LeBlanc and Zubeck, made the approach officially academic. [\[1\]](#)

The MDA Model in a nutshell

The MDA framework formalizes the consumption of games by breaking them into their distinct components:



...and establishing their design counterparts:



Pic 1: Goal of the MDA paper by Hunicke, LeBlanc & Zubek, 2004

The authors of the paper defined the eponymous parts of the MDA framework - Mechanics, Dynamics and Aesthetics - as follows:

Mechanics describes the particular components of the game, at the level of data representation and algorithms

Dynamics describes the run-time behavior of the mechanics acting on player inputs and each other's outputs over time.

Aesthetics describes the desirable emotional responses evoked in the player, when she interacts with the game system[2]

As far as I know, MDA was the first design framework that was adopted by both computer science and also at least a faction of the industry. Some developers really took to it, but it became a real hit in design schools and universities, where it is taught and welcomed by students to this day.

Part of that academic continuity, however, is predicated on the fact that the MDA model still stands exactly as it was erected by LeBlanc and his colleagues. And yet I'm pretty sure that's not what they intended when they first published the model. Instead, I believe they wanted to start a discussion -- to put the MDA model to the test, to fix it when it broke, to enhance it where necessary and ultimately to build a practical tool for the creation of games as *well* as an academic tool for criticism.

In the modern development environment, MDA not only fails to provide a framework for narrative design, there is no coherent interface. Is narrative design a component of mechanics? In part, yes, but hardly as a whole. It certainly doesn't belong to dynamics – and also not to aesthetics as defined by Hunicke and colleagues. Every attempt to apply MDA to narrative design rules stretches MDA to the breaking point.

The MDA model now is more than 11 years old, which is quite an achievement in an industry that reinvents itself every 2-3 years. Only recently, during the past few months, have we seen

articles in which the model has been deeply analyzed, criticized and challenged. The model has been discussed on Gamasutra by Luis Claudio Silveira Duarte[3], and an article by Lana Polansky[4] was followed by another by Frank Lantz[5], both making good comments on the topic. All of those voices in turn helped focus my own mind about MDA, and enabled me to better understand the sub-structures of game narratives.

I have nothing but the greatest respect for the pioneering work of Hunicke, LeBlanc and Zubek. But just like a Mercedes of today hardly resembles the first Benz motor car, we have to progress. There are problems with the MDA model – and the articles by Polansky and Lantz show that I am not the only one who wants to work *on* the framework so we can again work *with* it. We need such a model, and those articles do not discredit MDA or argue that a design framework is worthless. They show where MDA has weaknesses, and how we can overcome those weaknesses by fixing and enhancing the model.

Unfortunately, MDA will probably lose its acronym in the process because some of the biggest problems have to do with how the basic categories of MDA have been shaped. For reasons that will be made clear, in my own opinion MDA should evolve into DDE: Design/Dynamics/Experience.

90 % of MDA “mechanics” are not actually mechanical ...

In MDA, **Mechanics** describes the particular components of the game at the level of data representation and algorithms. One obvious problem with that approach is that’s it’s like describing an assembled car as “motor, gearbox and wheels“, when a car

is obviously much more. To the extent that “data representation” is mentioned it is only to a small degree “mechanical”. To a much greater degree it is “aesthetical”, because the data represented often consists of graphics or sound assets.

Worse, MDA terminology now begins to conflict with itself.

Wasn't “aesthetics” meant to be the emotional player response, which is something completely different from data representation? As Frank Lantz explained in his article,

“mechanics” in the MDA model basically means everything that is directly designed by the designer under no direct influence by the player: *“But even more problematic is the term “mechanics”. Again, MDA wants to use this word broadly to refer to all of the stuff that the designer has control over – not just the rules of the game but the materials as well, the recipe and the ingredients.”*^[6] I agree with Lantz's interpretation.

To drive the point home that the term *mechanics* doesn't really fit as a description, here is an incomplete list of things that the designer has full control over, which thus belong to the MDA category of “mechanics”:

- GAME CODE: Game Architecture, I/O, Game Objects, Game Rules (code level), Interface (code level), Interaction Design (code level) ...
- GAME RULES (documentation): Structure, Balancing, Timing, Space, Plot Branching, ...
- WORLD DESCRIPTION (documentation): World & Game Rules, Flora & Fauna, Societies, Characters, Religions, Laws, Physics ...

- STYLE (documentation): Graphics, Sound, Narrative, ...
- FUNCTIONAL INTERFACE (data representation): diegetic, non-diegetic, spatial, meta
- CONTENT INTERFACE (data representation): Interaction Design (interface level), Graphics & Sound, Narratives, ...[\[7\]](#)

As you can see, for a number of items on the list it's problematic to summarize them under the "mechanics" label. Some of the items aren't mechanical at all, like graphics style, which is clearly an aspect of aesthetics, yet is also under direct control of the designer before it becomes part of the final game experience. If we further argue that a graphical style guide is not represented on the level of data representation or algorithms, then the MDA model leaves no place for that crucial design decision – proving MDA to be incomplete as a framework.

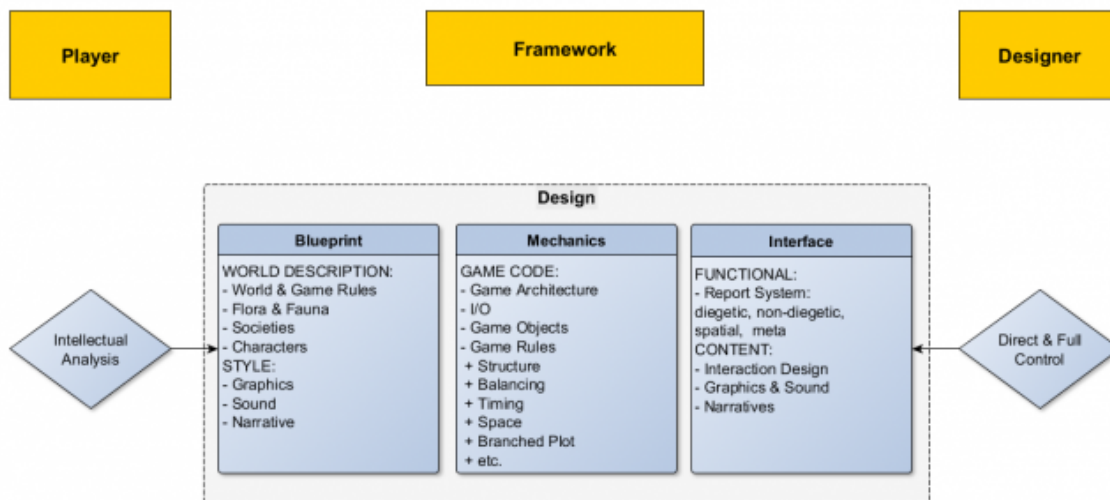
... but all of it is directly designed by the designers!

When I began trying to re-frame the MDA model I adopted two lenses. First, I wanted to focus on the production process of a game, especially with regard to its iterative nature. My version of the model needed to allow for that. The second lens came from a seemingly different angle that is in fact closely related. Why do game stories so often suck? I wanted a better understanding of narrative as an entity inside the MDA model – but it wasn't where I thought (and still think) it should be.

I am a producer and narrative designer by profession, so both of those lenses came naturally to me. As noted above I am writing a book about narrative design, and those two lenses also form the foundation of that book. But in order to actually build games

on that foundation I need a functional design framework in which narrative design can fulfil both its potential and obligations. My goal is thus to re-build the MDA framework so it will better support the notion of a journey for both lenses: the actual production of a game, and the eventual perceptive journey by the player. Yet no matter where I look, neither of those journeys are anywhere to be found in the MDA model.

While thinking about a re-framed “mechanics” category I came up with three sub-categories that allowed me to make clear distinctions without leaving blurry concepts lurking in-between.



Pic 2: The "Design" part of the DDE model

The above graph shows my take on what Zunicke and colleagues called “mechanics”, and what I would rather call “design”. Everything in that graph is directly designed by the designers, who have full control over that stage of the process.

(The term “design” was also proposed by Jesper Juul in an online discussion following Frank Lantz’s article, but I don’t know if Jesper would agree with my sub-structures.)

“Design” still has three important sub-categories, one of

which is “Mechanics”

The subcategories under Design are:

BLUEPRINT: Blueprint is manifested by that part of the design that deals with the game world *in concept*: its cultures, religions, physics and other rule sets; the free form notation of the game mechanics; and the developed styles of art design, narrative design, character design, and sound design. It is what the common understanding of “design” means: laying out ideas, painting, writing, inventing a world and its functionality, documenting it all in a Wiki and launching it into the iterative process necessary to become a solid foundation of the game itself. (For a while I used the term “setting” for this sub-structure, but blueprint now seems much clearer to me, since it reflects the dominance of planning and documentation throughout this development stage.)

MECHANICS: Mechanics remain in the framework, but are now much more specific. They include everything that will finally create the physical game *in the abstract*, meaning: in code. Mechanics are about the code architecture, the input/output handling, the object handling, the implementation of the game rules and object interaction, and other code-related stuff. It is what the player does not directly see or hear during play. Code remains invisible, makes no noise, and can only be experienced via the interface.

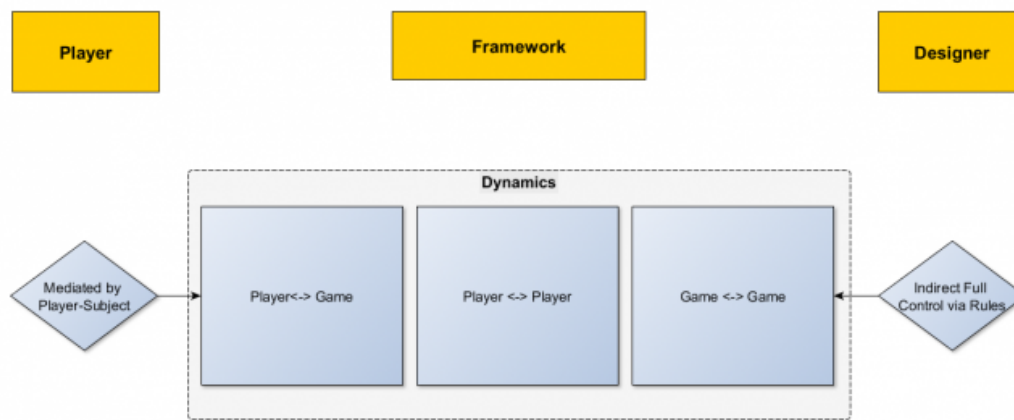
INTERFACE: Interface contains the design and production of everything that will finally create the physical game *in the concrete*, meaning: everything that serves to communicate the

game world to the player -- how it looks, how it sounds, how it reacts and interacts with the player and the game's feedback loops. Interface also contains the report system that every game needs, be it diegetic or non-diegetic, spatial or meta.[\[8\]](#) Thus every graphic asset, sound asset, cutscene or piece of text is part of the interface as long as it is also part of the game data. It is everything the player hears and sees, every piece of data that doesn't belong to the executable or configurative code level of the game.

One of the big achievements of the MDA framework: Recognizing the function of dynamics!

Building on the "design" category there were considerably fewer problems with the category of "Dynamics", though in practice we're still waiting for tools that will help us gain more control over that part of the creative process, thus saving us expensive design iterations. As Frank Lantz correctly noted: *"For me, the greatest strength of MDA is that it emphasizes the "second order" nature of game design. Mechanics is used to refer to the parts of the game that the designer has direct control over, aesthetics refers to the qualities of player experience that the game ultimately generates, and in between, linking the two, are the dynamics of the game in action – the behavior of the game's different parts interacting with each other and the player while the game is being played."* [\[9\]](#)

In the DDE version of the framework I extrapolate a bit on dynamics in order to add a little more detail, especially with regard to the different perspectives that designers and players have.



Pic 3: The "Dynamics" part of the DDE model

Returning to the Mercedes metaphor above, if “design” contains the planning for all of the parts of the assembled car, as well as the assembled parts themselves, “dynamics” defines what happens when you start the engine and all the assembled pieces work together: the pistons, crank shaft and valves of the motor, the gear box, suspension, the springing of the seats, the street, the weather, the driving style, the tremor of the steering wheel, the noises, even the song on the car stereo.

Note that this part of the design framework is *still* under full control by the designer – at least in theory, assuming an iterative design process. In practice there will always be too many different players to predict every behavior that each player might engage in. Thus the control of the designer is indirect, because all the designer can change throughout the iterative creation process falls into the category of “design” – where “dynamics” always adds a layer of emergence and unpredictability.

While the *player* is confronted with the dynamics of a game, that confrontation also is indirect. As Miguel Sicart has shown, the

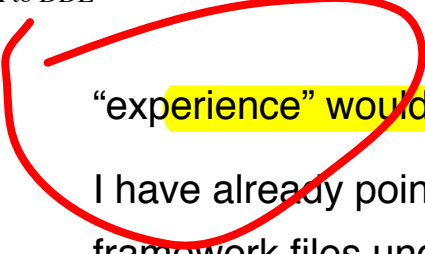
indirect nature of interaction happens via the creation of the player-subject.[\[10\]](#) Because that term will become important later, we will introduce it in some detail here:

The player-subject is based on the theory that it is not really *us* who are playing games, but subsets of ourselves. One can imagine the player-subject as similar to a mental avatar, a character existing solely in the mind that is able to make decisions we would never be able to make in real life. This is a character with a different set of abilities than we have – and a different set of ethics. In a lot of ways we put the player-subject into mentally dangerous situations in games to experience that danger without our real selves getting in harm's way. Thus we can experience and explore ethically and mentally challenging situations from a safe place. We can rake in all the benefits and rewards without risking the trauma that such real-life situations could cause.

In the context of dynamics, as well as the greater framework, the designer and the design thus do not deal with the player directly, but with the player-subject. It is that instance or aspect of the player that makes decisions while playing.

The term “Aesthetics” leaves out what game design really is: experience design

“Dynamics” is all the parts of a car in unified action, plus any external influences, but it is NOT the driving experience! Inside the original MDA framework that would be referred to as “aesthetics”, but there are a lot of reasons why “aesthetics” is an even worse term than “mechanics” – and why I believe that



“experience” would be better.

I have already pointed out that a lot of the stuff the MDA framework files under “mechanics” results from working on deeply aesthetical design questions. Thus calling a completely different category “aesthetics” would be questionable in itself. But the problem runs deeper:

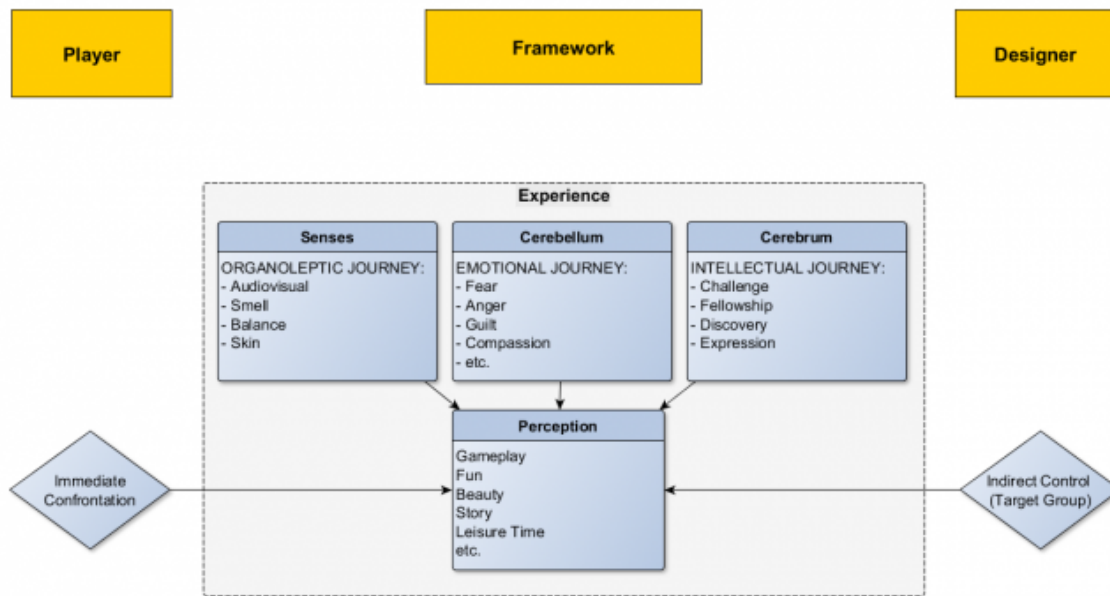
“Aesthetics” is a term used in philosophy in at least two ways:

1. It’s used in phenomenology, in trying to answer the question of HOW we perceive things. Kant for instance claimed that we cannot perceive anything independent from time and space, because these are two a priori intuitions we can’t help but have.
2. Aesthetical theory tries to answer the question of WHY we perceive something as beautiful or ugly, as art or non-art. What is it that creates beauty or art in our eyes and mind?

But “aesthetics” is also a psychological term, having to do with how different people perceive the same color, sound, melody, picture or text in completely different ways, as well as trying to understand the reasons and implications behind those differences. What is it in our mind that determines the volatility of our perception? And if that’s not confusing enough, “aesthetical” is also an everyday synonym for “beautiful” or “well-shaped”.

You can see how “aesthetics” as a term can easily become mixed up with the aesthetical side of the MDA design process. Even worse, the term is controversial even among philosophers, and its use among artists was famously denounced - not entirely without reason - by Barnett Newman, who quipped: “Aesthetics

is for the artist as ornithology is for the birds.”



Pic 4: The "Experience" part of the DDE model

Any practical game design framework should have at its core the goal of helping game designers understand how to approach their daily work. It should help frame the development process, while also accurately representing the underlying art form. A framework loaded with philosophical terms that are only useful for assessing art *after* it has been created is by definition of no practical use. If the DDE model can help the critics (and I'm sure it can), that's a bonus, but first and foremost DDE is for the artists.

In theory, of course, what we know from the various fields of aesthetics can help us make design decisions. In practice, a game often turns out to be too complex to plan detailed aesthetic effects in advance. But even the fact that knowledge of aesthetics can help us make design decisions is hardly sufficient for naming an entire sub-structure of a design framework after that murky word, particularly when the designer

does not have full control over its effect. Even worse: calling it “aesthetics” neglects everything that really IS central to that design category, which is “experience.”

Let’s recall the definition given by the MDA paper: “**Aesthetics** describes the desirable emotional responses evoked in the player, when she interacts with the game system.” Any reading of that definition involves time and space, in most cases over hours and hours of play. But for the participants something that is happening over time and in space is necessarily a journey, an experience. Game design is (or should be) experience design – and in fact that is mentioned in the MDA-paper: “In addition, thinking about the player encourages experience-driven (as opposed to feature-driven) design.”

As soon as there are dynamics, those dynamics can be experienced

That’s why I, in line with Jesper Juul, suggest changing the MDA category of “aesthetics” to “experience”. Experience starts as soon as the player plays the game. It is not separated in time from the dynamics, because dynamics also need time and space as *a priori intuitions*. When there are game dynamics, the player-subject can experience those dynamics. If the dynamics are sophisticated enough to create an antagonist for the player-subject, the game becomes a worthy opponent. As a result, playing the game over time will create a journey that works on many different levels:

Senses: The organoleptic journey consists of all the sensory experiences the player has from start to finish. It’s the totality of

what the player sees, hears and senses through the available output devices.

Cerebellum: The emotional journey consists of all the emotions the player experiences while playing the game. It's the emotional ride the player experiences, the fears and horrors, sadness, guilt and anger, and whatever other emotions the game evokes.

Cerebrum: The intellectual journey consists of all the challenges and decisions the player experiences and contemplates with the conscious mind.

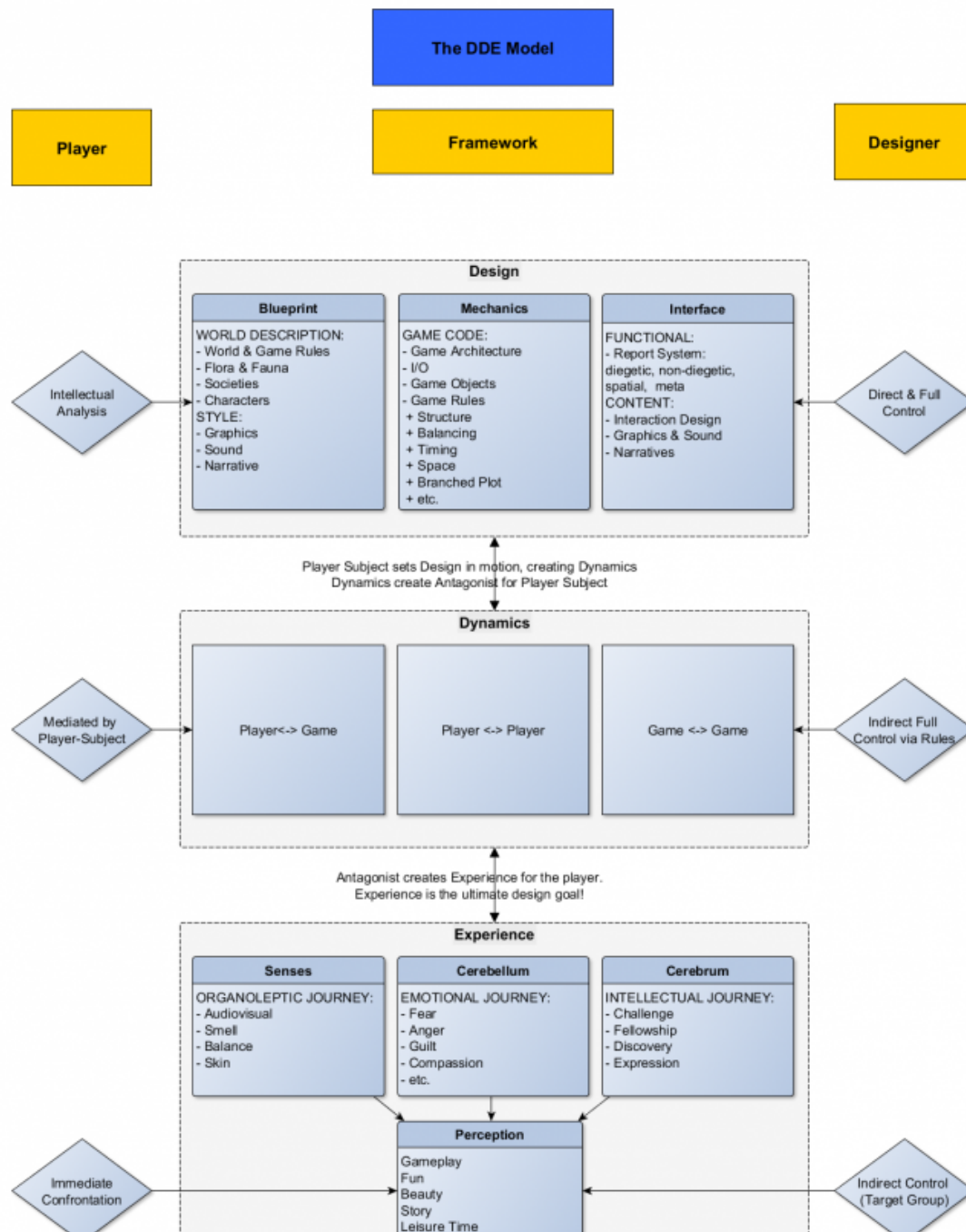
Together the three journeys will be formative for the player's *perception* of the game. This perception has many names and sub-concepts, and whatever the game experience is worth in various parts of the perceptive universe is a matter of personal taste and bias.

If we look at the role of the player we see that perception is the level of immediate confrontation. That is where all of the senses, emotions and intellectual capabilities are confronted directly – or would be if the player did not create the protective layer of a player-subject.

From the perspective of the designers, that is also the level where we will lose full control over the work (even indirectly), because that is where the individual perception of the player takes over. During the player's experience of the game, the player is no longer restricted by the code. The player can sense, feel and think whatever they want about the experience. They can like it or hate it, enjoy it or find it boring, get what they

wanted or didn't expect, be challenged or taken to the emotional brink.

The better the designer knows the target audience, and the better the game targets that audience, the better the game will be received. That is why objective criteria can only take design so far – and why knowing the expectations of your audience is critical to all measures of success.





Pic 5: The whole DDE Model

With the DDE subcategories in place, it's time to define the relationship between the different parts.

From the original MDA framework: *“Each component of the MDA framework can be thought of as a ‘lens’ or a ‘view’ of the game - separate, but causally linked. [LeBlanc, 2004b]. From the designer’s perspective, the mechanics give rise to dynamic system behavior, which in turn leads to particular aesthetic experiences. From the player’s perspective, aesthetics set the tone, which is born out in observable dynamics and eventually, operable mechanics.”*[\[11\]](#)

Apart from the terminology used the sequencing is obviously correct, and the DDE system supports that explanation if we simply replace the old terms with the new ones. But there is a danger of misconception underlying the entire premise, because that sequencing does *not* describe a timeline. Instead, everything in that design sequence can happen at the same moment.

For designers that is the description of an iterative process, where every decision on every level can change the whole game. For the player that is what they are doing and experience when playing a game. The player senses the world *and* analyzes the underlying mechanics at the same time, from start to finish, confronting and experiencing every aspect of the design. (In practice designers are also the first players – or at least should be.) And that’s why we need to add one more term

in order to strengthen a specific perspective: the *game as a confrontation* over time and space; a set of challenges that can create a wide range of perception journeys, depending on what the designer has created and how the player understands it.

A good game must be a good antagonist for the player-subject

Since we renamed the last category “experience”, at this juncture it makes sense to describe the entire framework more from the player’s angle than from the designer’s point of view. Ultimately the designer has to understand the perspective of the gamer to make a great game, and at the risk of repeating myself, **game design IS experience design!**

That’s where the concept of the “antagonist” comes into play. The antagonist is well-known from storytelling and other narrative art forms. In different guises the antagonist exists in basically every art form, be it counterpoint in music or complementary colors in painting. And the reason is obvious: it is through conflict, or contrast, or tension, that almost all art generates interest at differing levels of awareness.

If a designer thinks of the dynamics of a game system as the antagonist of the player-subject, that perspective allows the designer to comprehend the whole experience of playing a game as a narrative. Yes, there are certain expectations we have about narratives, and good designers will anticipate those expectations when creating a great narrative experience. If the antagonist of a narrative doesn’t fulfil that role, the narrative will be boring, or at least less interesting than it might have been,

and it may even fall apart.

In talking about narrative we do not only mean any embedded story that the designer creates for the game.[\[12\]](#) Of course any embedded narrative (including deciding whether one should be included at all) should support and reinforce the complete experience. If the perception journey and the embedded narrative don't match we get what Clint Hocking has called the "ludo-narrative disconnect"[\[13\]](#).

When played, every game creates an over-arching meta-story, an emergent story. But in order to do that successfully the game must become the worthy antagonist of the player-subject. And just like the antagonist of every story is responsible for the set of challenges the hero has to overcome, the antagonist that is the game system is responsible for the obstacles the player will encounter and overcome (or not) while playing.

Even if you prefer different terminology, I think we can all agree that a well-defined design framework should show that connection. If the game is not the right challenge for the gamer -- if it's not a worthy antagonist -- the player will stop playing, or continue playing but be underwhelmed or even bored. The DDE framework supports the notion of conflict from abstract to literal; MDA doesn't mention it at all.

Design, dynamics and experience are not production stages

Returning to the designer's (or producer's) point of view, again, the three sub-structures of the DDE framework are not to be interpreted as production stages. They are, rather, what one

must understand in order to set up an effective iterative process for creating a game.

As noted above, the first players of any game will usually be the developers themselves. They iterate the design through change after change until the dynamics begin to produce the desired effects. They are the first ones to perceive the game and to get an idea of the journey it will produce for the audience. With each iteration, every single line of code and every single asset, whether it be graphics, sound or narrative, can be adjusted to alter the effect that the game has as an experience.

Iterative production cycles today should be a given, but in my experience they are the first thing to go when time and money are running out. Unfortunately, without them you *cannot* develop a good game. A thousand-piece jigsaw puzzle thrown in the air is never going to put itself together. And games usually have considerably more than a thousand pieces.

The DDE model allows designers to assess and re-assess the value of story within the overall framework of game development. Look again at the entries in the blueprint substructure and you'll see that nearly all of them have a narrative connotation. And as a first consequence of that, it is long past time that narrative design be considered an integral part of any design process from day 1. (The persistent disregard for narrative design is an artifact of commercial conventions and power dynamics which have nothing to do with MDA or academic criticism.)

The DDE framework is obviously not finished. So far it serves the purpose it was created for, which is laying out a stable

foundation for the things I want to talk about in my book. As time passes and I think about it more deeply the framework will inevitably undergo more changes. Some of the terminology may change, some definitions will get more attention, and there are still a few areas I have yet to fully explore.

Still, it has become sufficiently stable that I am now looking for feedback from any quarter, because I know I can't finish it on my own. There are too many fields involved where I simply don't have the expertise. So it's time for other voices and other lenses to come into play. It's time we try to rip this and every other framework to pieces to see if there's anything of worth.

Let's not wait another eleven years.

(Since I'm not a native speaker this blog post has been edited for correct English by Mark L. Barrett, a long-time friend and brother in arms. I owe you, Mark! Please visit Mark's blog "The Ditchwalk", where you can find lots of insightful posts about interactive storytelling: <http://www.ditchwalk.com/category/interactive/>)

[2] l.c., p2.

[7] For me everything that is displayed on screen or can be heard through the speakers is part of the game's interface, because it translates the abstract layer of code into something the player can understand more easily. This has certain implications as you will notice later in the article.

[10] Miguel Sicart: The Ethics of Computer Games,

Cambridge/London, 2009.