**Peter Ciszek**

# 3D Production Pipeline in Game Development

Bachelor's Thesis
in Information Technology
December 8, 2012

**University of Jyväskylä**

**Department of Mathematical Information Technology**

**Jyväskylä**

**Author:** Peter Ciszek
**Contact information:** peter.m.ciszek@jyu.fi
**Title:** 3D Production Pipeline in Game Development
**Työn nimi:** 3D-tuotantolinja pelikehityksessä
**Project:** Bachelor's Thesis in Information Technology
**Page count:** 20
**Abstract:** 3D production has grown to be major part of the digital industries such as game and film industry. In these industries work processes can be described as production pipeline. Pipelines are constructed from actors working on the pipeline, their workflow and tools used to do the work. These pipelines have been used to make processes more efficient and rational. This study attempts to explain the meaning of 3D production pipeline for game development and evaluate the processes. In this thesis, we examine the topic by examining the existing domain literature and by describing a simple model of a production pipeline.
**Suomenkielinen tiivistelmä:** 3D-tuotanto on merkittävä osa esimerkiksi peli- ja elokuvateollisuutta, jotka ovat itsessään suuria teollisuudenaloja. Näillä aloilla työprosesseja voidaan kuvata tuotantolinjastoina. 3D-tuotantolinjastot rakentuvat sillä työskentelevistä tekijöistä, näiden työnkuluista ja tässä työssä käytettävistä työkaluista. Tuotantolinjastoillaon tehostettu ja järkeistetty toimintaa. Tämän tutkielman tavoitteena on selvittää 3D-tuotantolinjojen merkitys pelikehitykselle ja tarkastella siihen kuuluvaa tuotantoprosessia. Asiaa tarkastellaan tässä tutkielmassa tekemällä katsaus alan kirjallisuuteen sekä kuvaamalla yksinkertainen malli tuotantolinjastoille.
**Keywords:** 3D production, pipeline, 3D modeling, game development
**Avainsanat:** 3D-tuotanto, linjasto, 3D-mallintaminen, pelikehitys

# Contents

# 1 Introduction

Game industry has grown huge and shows no signs of relenting concerning the long term growth. [15] [5] When one explores games published recently, it is obvious that 3D production is an essential component in many products. For example in 2010, a survey for indie game developers found out that 41% of game development activities are art related, which includes 3D production as well [20].

As a discipline, 3D production has accumulated different tools and methods to answer the changes from rapidly progressing technology. Continuous increase in computing performance makes it possible to create game content in new ways. [17] Production and work methods progress to answer these new ways and their requirements. [26]

This paper attempts to show briefly how production pipeline is structured for 3D production in game development. We will also describe the game development as process and take a look at its organizational structure. To understand the phenomenon and the work environment, we will discuss the actors involved in production, as well as their actions.

This thesis is structured as follows. In Section 2 we go through the game development process and in Section 3 we discuss 3D production in games. Section 4 describes a production pipeline, and shows how it is built and why. Finally in Section 5 we conclude.

# 2 Game Development

To understand the usage 3D production pipelines in game industry, we need to take a look at the process of creating games. Creating high-profile games in large organisation is challenging and requires coordination. [21] Because of this, game development process is divided in project phases. In these phases there are formal procedures, roles, and deliverables.

## 2.1 Disciplines

In the early days of game development, the disciplines overlapped more because the work of designer, programmer, and content creator could be done by a single person [3]. When industry evolved and got larger, the need for domain focused specialists arose.

These specialists are normally identified by their discipline, and those disciplines

are defined by the work domain within the game development process. In his article *The Disciplines*, Albert Ferrer categorizes the game development discipline by actors: game designer, programmer, artists, producer, audio designer, quality assurance tester, and other jobs [7]. His description cover all the major disciplines in the industry, and this taxonomy also correlates with domain literature examined in this paper. Also note that the divisions and departments of game development organisation are usually referred to by the corresponding discipline name. [18].

As a compilation from literature, this paper will cover game development process from the viewpoint of four domain: business, technical, design, and content. However, this is not intended to underrate the significance of the other disciplines, but to abstract the environment for the description of 3D production pipelines. The main reason for this taxonomy is to simplify the process to be able to focus on the actions on the pipeline.

### 2.2 Project Phases

A game development process is normally divided into three phases: pre-production, production and post-production [15]. This phasing affects the 3D content production and this relation is examined in detail in Section 3. The *Desinging virtual worlds* by Richard Bartle [2] describes virtual world development process with four phases: pre-production, production, roll-out and operation. In this paper, we use less game-specific partitioning which is also more common in game development literature, such as: *Basics of Game Desing* by Michel E. Moore [15] and *Better Game Characters by Design* by Katherine Isbister [10].

In game development, the amount of work varies in different phases. Figure 1 presents a simple overview of a game development process. The figure presented is adapted to the same phasing that we use in this paper. The coloured lines indicate the required work or workforce during the process. We see that the amount of work during the process is not static, and the work load is occasionally uneven. It should also be noted that the graphs change depending of the type and magnitude of the development process.

It is not uncommon for large companies to have multiple development processes under way for different product titles. Inside these organisations, actors may be moved between different projects, for example, to ease work load of other project or to counter project delays.

In this paper, we are not interested much of the general business or management side of the game development process. However it should be noted that schedules

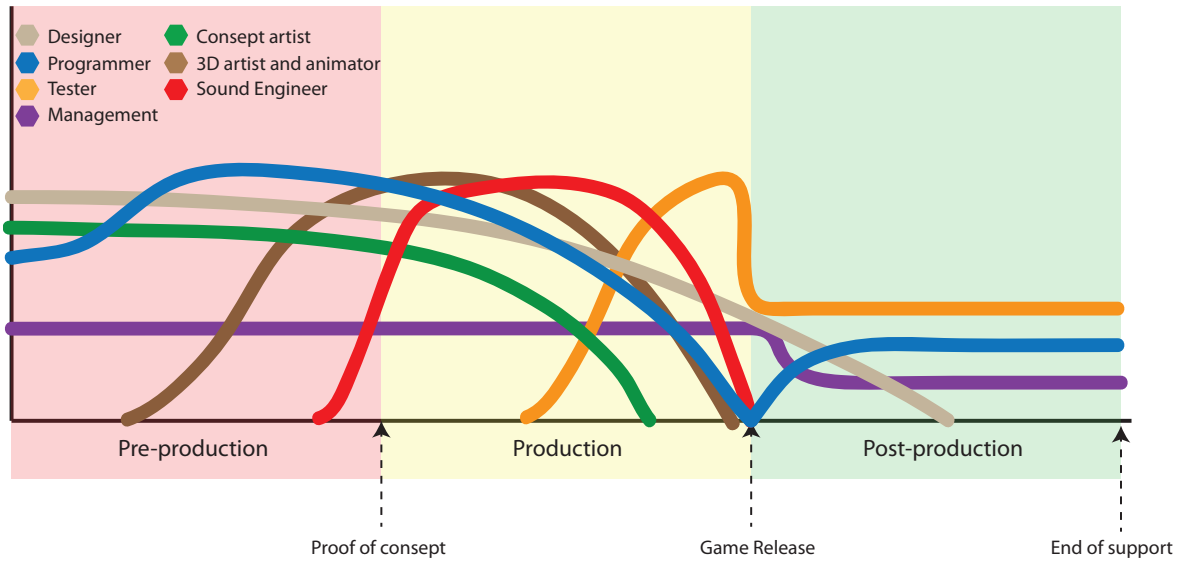


Figure 1: A simple overview of some game development process. [23]

and business and marketing plans are produced in the pre-production phase.

Pre-production phase can be seen as a overall planning and proofing phase and in this phase the overall vision of the game is communicated [2]. This phase is also where the production tools is supposed to be initialized [8]. In [8] Patnode points out that it is important to have solid and working pipeline ready before the production. In the end of pre-production phase prototype is produced and the project proceed to production phase [2].

Production phase is a central part of the game development and all the requirements should be clear before it begins. When the production phase starts, it is expected that every division, team, or domain specialist know what to do, in general, and how. In production phase, it is also expected that the production tools are set up and tested for the content creators to proceed in create the expected content. The Production phase is longest part of the game development and can take several years. The phase ends to the release of the game. [15]

Post-production is the last part of the production process, which basically means game support. It is normal that published games contain defects that needs to be fixed by patching. [15] It is also usual that games with multiplay or online features have to undergo game design tweaking and content fixes after the release. Fixable content, for example, being a faulty animation or a sound with inadequate quality.

It should also mentioned that the film industry shares the same kind of phasing with the difference that the product is not finished by the end of the production

phase but after the post-production [6]. This is important to keep in mind as a lot of the material concerning 3D production and pipelining is written for the animation film and visual effects industries.

As a whole, however, game development is first and foremost a software development process [3]. All the practises and standards for good and stable software development apply. This is crucial because the technical environment and platform frame the content creation process. Basically this means that the content creator's main function is to create the required content based on the design, while taking account the given technical requirements and restrictions.

It should noted that the pre-production phase can also be described as a requirements gathering and these requirements are called deliverables. For example, Bartle describes pre-production phase through these deliverables. This deliverable-based viewpoint is fitting because when discussing the work on a pipeline, we are usually interested in the requirements for the work at hand and what the output should be.

### 2.3 Deliverables

When working on a video game in larger organisation, one has to have an understanding on what needs to be done in one's work domain during the development process. Some of this knowledge can be obtained from the requirements engineering subprocress of the software development process, especially when we discuss about technical side of the game development. However, when talking about games there is rarely a customer or client with a specific need for a piece of software whose functionality can be described in detail to generate the requirements.

The requirements for the game are gathered by developing a set of documents, or presentations, that, in this paper, are called deliverables. These deliverables can be seen as something that are needed to clarify and communicate, or to deliver, the requirements. In Figure 2, we can see how the preproduction phase advances from the perspective of these deliverables. Sometimes these items have dependencies, which means that one needs to be completed before the production of another. For example, it is obvious that without reading vision document, it is challenging to comprehend what systems need to be defined in the technical document.

Not all the deliverables are mandatory in every game production, and depending how the organisation has defined them, some deliverables could contain similar material to others. It should be noted that we found no reference to using pipeline document as deliverable, but sake of the topic of this paper we present it as one, and this can be seen in Figure 2. Next we will go through the deliverables that will affect

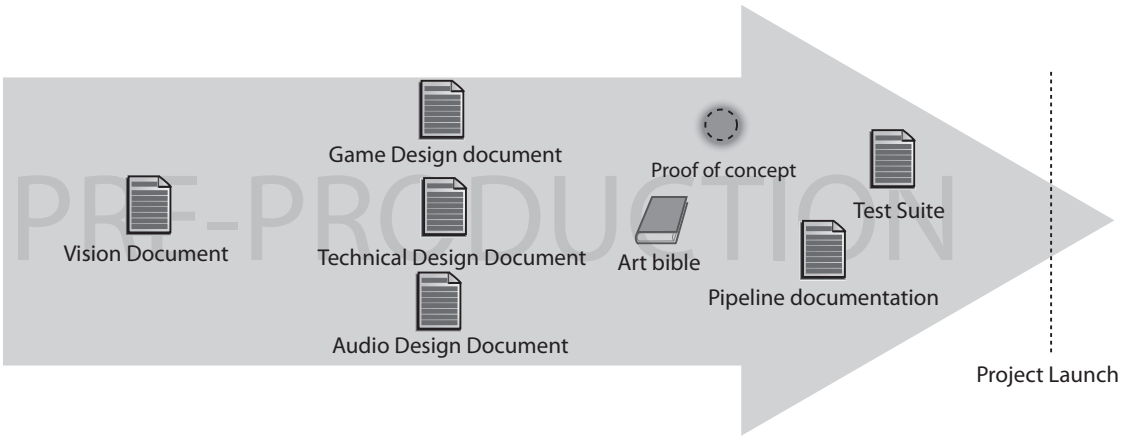


Figure 2: Pre-production phase with deliverables.

the planning and building the pipeline.

**Vision document** is a few pages long description of the feel and style of the game. This is normally first document to be produced. Vision document has also other names such as pitch paper [15]. Typically, the visionary or "idea man" formalizes this document to be presented for the management or the investors [15].

To contain the reference material for content creation, an **art bible** created during the process. This document is primary used to keep the content coherent [19]. Art bible is also created in pre-production phase [15] [2]. "[An art bible] is often composed primarily of concept sketches and other resources that artists can refer to as they are working on creating various visual assets for the game" [19]. The art bible is not typically finished during the pre-production phase as complementary material is added during the project.

**Designs documents** such as game design document define how the game works and what is the gameplay and technical design document contain the requirements for the game engine as well as architectural description of the system [3]. Programmers and artists might assist in creating this document [15] [2]. Requirements for music and sounds are described in the audio design document [15]. The technical design document is created by programmers to define all the work in the development process for software engineering side. Development teams or departments may need specific tools to be able to perform their job. These tools are also described in the technical design document [15].

**Proof of concept** or **prototype** is created before launching the project and it clarifies the workflows for the development process [15]. **Pipeline documentation** is finalised after the pipeline is tested in practice, this is done most likely with the pro-

totype. Quality assurance creates document called **test suite** based on the design documents. The test suite describes what should be tested and how [15].

The project deliverables present the knowledge what to do during the development processes and directs how it is done. A more detailed example of these items can be found at [19].

### 2.4 Actors in Content Production

Game development requires usually professionals of different fields [3]. In this paper we call these domain professionals *actors*. This "actors and actions point" of view have also been used in [22].

Actors, roles, and positions are not synonyms. To avoid misconception we would like to underline that an actor is substitute word for an individual, a person and a human being with a personality, who holds different skills and aptitudes. Role, on the other hand, present a distinguishable set of actions that defines that role. For example, role of an animator may contain acts of rigging and animating. Positions then are more or less like job positions that can contain many roles. It should be kept in mind that organisations have developed their own work process, and the positions and roles in different organisations vary.

We will briefly go through the actors working with the pipeline. **Director** is an experienced content creator or manager who manages the content creator team as well as ensures the quality and design consistency of the content products [3]. Director also takes central part in the planning and building of the production pipeline, thus needing the experience from the workflows [21]. **Designer** generates requirements to the content creators by contributing to deliverables. **3D content creator** uses 3D computer graphics software to tools to create 3D content and **2D content creator** then uses traditional and digital methods to produce digital visual assets and textures.

### 2.5 Development Process

In short, game development can be described as a process that will produce a game or a video game as its output. In this paper, we discuss products that contains 3D content and thus 3D production is needed. Of course, games can also be made without 3D content or to have trivial amount of it.

Studies and literature indicates that game development is not straightforward. Bruce R. Maxim and Benjamin Ridgway in their publication *Use of Interdisciplinary Teams in Game Development*, made an incisive portrayal of game development. They

write: "The development of computer games is labor-intensive. Modern game development requires the effort of a team of skilled professionals to integrate multimedia content with complex software. Producing high-quality software game products requires large teams to rely on high levels of communication, organization, and planning to avoid costly delays and failures." [14] In the context of this thesis, this refers especially to the integration of multimedia content, communication, organization, planning and avoiding cost, delay, and project failures. Some of these elements are discussed by Bethke in [3] as the things that make game development difficult.

In fact, having a successful development process is hard. High failure rates in production are extremely weary for the developers and in long run will, most likely, bring the company down [21]. The fact that in the beginning of this century only one in five new businesses succeed, gives a harsh image of the industry [13]. This is true especially when we consider that factors increasing the failures have not disappeared but the volume of work and workforce have considerably increased [21].

The development process starts from an idea or a vision. This idea is normally presented or proposed to some body that is expected to fund or approve it. This body or organisation possess actors with needed expertise to produce a video game. These actors, normally with the person that presented the idea, start working on the details of proposed idea. In short, the key figures who propose the game, produce documentation that defines the requirements for the whole project. These details are delivered in pre-production phase through deliverables. These deliverables consist of all the requirements and instructions that can described in that point of development. During this pre-production phase, proof of concept or prototype is produced to ensure that the production is plausible.

When all the mentioned deliverables have been produced, the project is launched. In this point the actors start working on the game in teams and departments specified by their discipline. Then, in the better part of the production phase, test versions of the game are done and put in the testing.

The production phase ends in the release of the game. This means that the product is ready to be packaged and delivered to the customers or retail sellers. The game development process ends in cutting of the support in the post-production phase. This means that no one is any longer officially working on the game in form of patches, fixes, or support service.

# 3 3D Production

One major area in creating high profile games and animations is the production of their content. Nowadays this content is mostly created in three dimensional format. The process of producing this digital three dimensional data is here referred to as *3D production*.

## 3.1 3D Production in Game Development

3D production is significant part of game development because most of the high profile games nowadays are, more or less, 3D games. These games are developed by large organisations that have number of studios under them and these studios, have departments and teams for 3D production. If we take a look at any game these organisations have produced, we can see that a large fraction of the positions credited relate to content creation.

There are various names for describing the discipline of creating 3D content. International Game Developer Association (IGDA) uses term visual desing when describing disciplines of game development. IGDA describes the Visual design discipline as "[the act of] analyzing, designing, and creating visual game assets." Naming is not entirely established and new terms and categories might still be proposed in the field. The naming issue and debate is also noted in IGDA curriculum framework. [9]

In this paper we use term *content creation* to describe the activity to create any game content. This includes creating all the sounds, music, effects, voice acting, motion-capture, animation, models, and in-game objects and environments.

As a part of content creation, *3D content creation* refers to actions or work, which results in a product with three-dimensional data or a resource that is used with 3D data. With same idea, 3D content refers to a product that contains 3D data or resource. Examples of such resources include texture map and motion capture data. We define here that any material or data that cannot be sensibly used alone or without three-dimensional also data falls in this category.

3D content in games is commonly created for real-time rendering [26], in which the deliverables and requirements for comes from the game engine, platform, and hardware. There might also be several different techniques and middlewares used in the production that creates special requirements for the work at hand. For 3D production, these requirements normally work as guidelines and boundaries on how the 3D content should be created. Also, engines may come with specific tools that is used to create assets or content beside the traditional 3D production. In short, the re-

quirements for 3D production in game development are defined by the technologies used in the development process

These requirements direct what type of action are needed when 3D content is created. In next section we will briefly go throught some of the actions used in 3D production during the game development.

### 3.2 Actions in 3D Production

In 3D production the product is produced through series of actions where each action adds something to the product. Each action thus builds on the previous stage, forming the process of creating the 3D product.

The typical process of creating a 3D product can be desciribed as follows. First, visual assets are created by **concept drawing** [6], which are then are used as a concept for the modeling. **Modeling** is then used to create a mesh model with 3D modeling software [24]. After the model has been created, a process called **sculpting** can be used to add details to the model [11]. The model is then **textured**, that is, visual assets are created for it, which define the color of the surface of the otherwise blank clay statue-like model [24]. In addition, **UV-mapping** is used to define how these texture maps are positioned on the model's surface [24]. If needed, the model can be **rigged** for movement, which means creating a set of "bones" for the model and connecting the bones to the model's mesh [24]. Finally, **animating** is the action where the movement for the bones is defined and recorded [24].

Of the mentioned process, modeling can be seen as the main activity in 3D production. There are two main ways to create 3D models: surface modeling and solid modeling. In the game industry, most of the assets are created as surface models. This is because they are much easier to handle by the modelers and the game engine. Solid modeling is mostly used in simulation [1] as this reflect the real object more accurately.

The different actions in 3D production are done with software tools. Today the industry is in a state where most of the actions needed to create the 3D product can be done with single piece of software. Examples of this type of products include 3ds Max [24] and Maya [6] [12]. In addition, specialized tools for each action have been developed. These surpass the generic tools in many cases in their productivity in the specific action. For this reason, it is common that companies use different tools for different activities: one for modeling and another for sculpting etc. An example of this type of sculpting tool is Mudbox [11].

### 3.3 Workflows and Best Practices

In general, workflow is a set of specific actions that produce an output. To understand what is an workflow in 3D production, one may simply think any set of instructions, such as tutorial, that advices how to do create specific 3D content. We can be perceive workflow as a set of things that must be done to achieve some expected result. These workflows can contain reminders or tips that advice the actor to use certain methods during the work. We call these best practices. For example, creating models in modular manner by using specific modifier in the process, may be seen as best practice. In short, best practices are instructions guiding the workflow.

Beside the best practices, there are also common rules for creating real-time content, such as model topology cleanness and optimization [1]. These rules are gathered from the deliverables to confine actions in the 3D production. One of these is, for example, the polycount, which defines the maximum polygonal detail for the product. Polycount is used to ensure that stable frame rate is achieved when models are rendered in real-time. This is because calculations need to be fast in real-time rendering to get the required smooth sixty frames per second presentation [1].

The need for rules and best practices may arise on some occasions such as in-game content creation. For example, the actor responsible for this content might create surface that has two visible faces top of each other within close proximity. And when the content is used in-game, the bottom face flicker through the one on the top, even there should be several units of space between them. This phenomenon is know as Z-fighting and it is caused by the precision limits of the Z-depth of the objects [1]. The term comes from the flickering that in a sense is caused by two surfaces fighting which one gets to be on top and drawn.

One of the challenges in 3D production is the quality of the end product. If we ignore the aesthetics and discuss just the technical side, it is crucial to understand that the work field contains unexpected fault elements. The z-fighting is an example of such element, and content creators in long term just have learn to work around the issues. This includes the ability to use the given texture side limits and to understand the different mapping techniques. [1]

New workflows and best practices arise because process and work methods evolve to answer new challenges and requirements of the industry [26]. Utilizing new technologies and using them to streamline the workflows, is a challenge for whole game development organisation.

# 4 Production Pipeline

Management of the work, people, and tools is needed during the 3D production in the game development process. The production pipeline offers an method, or an interface, to manage, develop, and design those three as a whole, single construct.

## 4.1 Need for Managing Workflows

Workflow maangement is needed in content creation. As opposed to not manage the workflow in general, there are benefits for managing a workflow which we will briefly examine in this section.

First, one of the reasons is that using a workflow management will save working hours during production [8] [12] [16]. Also, the workflow management makes those hours more profitable. Patel argues in [16] that it makes possible expand the range of service offered by increasing the capacity.

Stacey and Nandhakumar also argue in [22] that game development process contains temporal inner and outer working rhythms that project manager need take into account. Workflow management makes it possible that the people working on it may proceed in different pace, because workflow management does not specify timing. In other words, workflow management makes it possible for multiple people work on same model in different phase of workflow, thus making it possible for the work to happen concurrently [8].

There is also a competitive reason for workflow management. When using off the shelf product, one is tied in competition with other companies and a well managed workflow makes it possible to have a competitive advance by being able to produce more in less time [16]. When it comes to content creation, we know that almost all the state of art tools are also available other companies in the industry and workforce is drawn from same pool of educational bodies. In other words, better practices during the projects give competitive advantage.

Based on the observations given in the literature, we can conclude that using a proper workflow management strengthens the workflow and promotes synergy inside it. However, we recognize that if the utilizing the workflow management fails, it can be hindrance for actors on the workflow management, as well as for the management. This simply means that management cannot be too rigid or formal.

### 4.2 Managing Production using Pipelines

Workflow management is usually discussed in the industry as *a pipeline*. Using and building a pipeline is, first and foremost, a form of management. However, having a pipeline does not do the management and work for you, it just makes it more clear.

In [8], Jason Patnode described a pipeline simply as "... a path that assets travel to make it into a movie or game." Tom Painter, in that same book uses description such as: "The way we work, the tools we use, and the order in which we do things are known as our *pipeline*." Also a definition for a pipeline were presented in [25] which was simple as: "A well-defined set of processes for achieving a certain result."

Beside the definitions in the literature, Mayur Pater describes in [16] the production pipeline as a dataflow, a workflow, and tools that facilitate those two. He defines the workflow as a process that regulate the work and the dataflow as regulation of data. In this case data consist of the information moving between people and the information needed by the people. In his description tools are used to improve these two flows. [16]

Finally, most precise conceptual definition we found for the production pipelines is defined by Dane Bettis in his thesis [4]. He describes that that production pipeline is not a structure, but more in sense of adaptable set components and this set then can be arranged and configured anew when needed. Those components are departments or teams which have their own task domains and digital hardware and software systems. In short, his thesis describes digital production pipeline as adaptable systems and structures built from synergy of human resources and tools. [4]

Pipeline describes the production process, making possible to examine how the actual work is done before and during the production. Production pipeline enables of building manageable production environment for 3D production.

### 4.3 Building a Pipeline

Pipeline building begins by gathering resources for 3D production. This simply means going through actors, tools, and other assets that will be available during the process. After this the deliverables are reviewed to evaluate whether these resources are sufficient for creating the required 3D content. This reviewing may involve the actors, as domain experts, to explain the workflows and capabilities of the tools. In the end of this process new resources, such as actor and tools, need to be acquired.

Based on the conducted review, the builder of the pipeline lists the required workflows, as well as the assets they produce. Next he specifies the actions inside the workflows, and the tool where those action are preferred to be executed. Then he

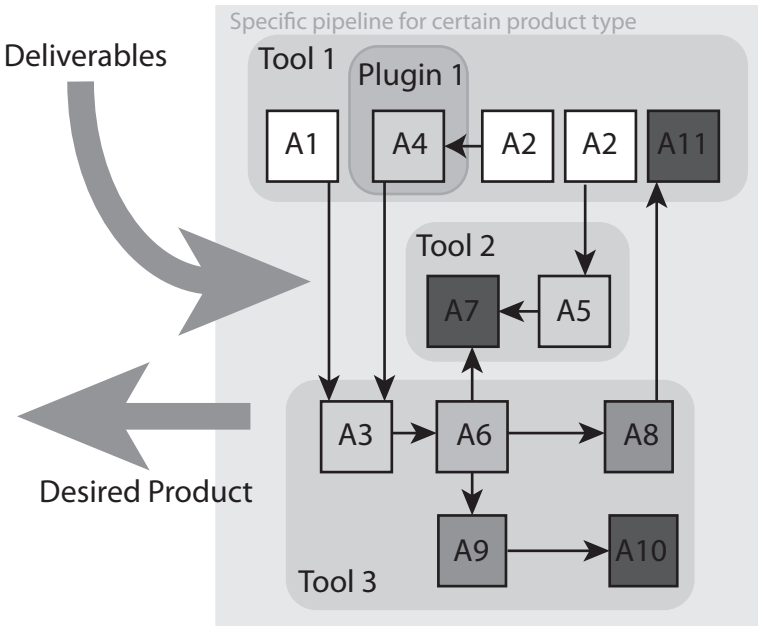


Figure 3: A specific pipeline for certain content product.

sets those actions in order based on the dependencies between the produced assets.

Building a pipeline also involves planning the infrastructure for the tools and assets during the production, as well as overall communication. This basically involves setting up data storages, communication methods, and other rules that are common in project work in software development.

The literature referenced in this paper also contained advices concerning building and managing pipelines. First, building a pipeline based on expectations is an error. Patel warns that the pipeline designers draw the pipeline according what they want to happen, instead what is going to happen or is already happening [16].

Also, when building a pipeline, it should be planned how to get information out of it [16]. This means ways to oversee the workflows and keep contact on how the actors work on the pipeline. From this we observe that pipelines should be build based on the capabilities of the actors. Actors skills and aptitudes should be one of the factors to consider when building a pipeline [16]. We discussed briefly the actors and the factors concerning them in Section 2.4. Again, only real assets and actor capabilities should be taken in account when building a pipeline, instead of expectations. This involves integrating efficient personal workflows for the actors on the pipeline.

When discussing on actors, it should be pointed out that attaching them to a process should be avoided. Patel argues in [16], that if processes are being contin-

uously related to certain person, a "gate keeper" is created. This means that one of the key attributes of pipeline, the change of roles, is eliminated by creating a bottleneck actor. Patel notes that when designing a pipeline one should keep the actors and processes apart from each other and distribute roles widely. This however, does not mean that *processes* and *roles* should kept separated [16].

In Figure 3, we present our pipeline illustration by using actions as building blocks and then by adding these blocks inside the tools used in the pipeline. Actions are indicated with identifiers A1...A11. In this figure, we have purposely left out the naming to keep the focus on the actions. Early stage actions are lighter in the picture and the color goes darker when we move forward in the pipeline. We did not include assets, roles, or positions to the figure, to highlight the fact that when building a pipeline, the focus should be more on the work and processes than data [16].

In the end, we would like to point out that before approving the pipeline it should be tested. In practice, this means that some of the most common cases of content desired in the project, should be ran through the pipeline. One option is to test the pipeline when building the product prototype, because also the prototype needs content to be plausible.

## 5 Conclusion

As a part of content creation domain, 3D production is significant part of the game development industry which undergoes constant changes, caused by the technological and methodical advancements. Studying the topic was not straightforward, because taxonomies change depending sources in domain literature. Companies create roles and positions for their own needs making generic classification difficult. However, we found out that there are disciplines that are overall accepted to describe the work done in the industry.

We found out that the game development takes a lot of time, people, and money. There are uncertainties in the process and mistakes are made. Cost of mistakes can slow down the development process and even cause the projects to fail. Game development process is first and foremost a software development process that is done in three production phases and it ends in cutting the support from the game product. During the development phases the amount of work is not static.

3D production in game development means creating 3D content and assets that are used in the game. Design based decisions are normally made in the beginning of the development process by domain experts and communicated through a set of

documents. Workflows and data flows inside the domain are described by a structure called a pipeline, which is built based on this documentation. This documentation describes the workflows and data flows, and is used to build the pipeline.

We found that there is already an accepted definition for production pipeline. A pipeline is a structure of tools, actors, actions and synergy of those. The 3D production pipeline is also used to manage and plan the 3D content creation process. In short, the pipeline describes how the work, in this case 3D production in game development, is done inside an organisation.

Production pipeline is not a tool for having a complete access to the work- and dataflows, but more of an interface to manage what work or actions should be essential for achieving product under the given requirements. Of course, pipelines may also be misused and badly utilized pipelines may affect the capability of the actors and cause undesired or faulty end products.

In the end, we can conclude that having a managed 3D production pipeline in game development eases the production. Pipeline clarifies the processes and also works as a way to communicate for work at hand.

## References


[1] Tomas Akenine-Möller, Eric Haines and Naty Hoffman, *Real-Time Rendering*, Third Edition, A K Peters, Ldt. Natics, Massachusetts, 2008.

[2] Richard A. Bartle, *Designing Virtual Worlds*, New Riders Publishing, USA, 2004.

[3] Erik Bethke, *Game Development and Production*, Wordware Publishing, Inc, Plano, Texas, USA, 2003.

[4] Dane Edward Bettis, *Digital production pipelines: Examining structures and methods in the computer effects industry*, Texas A&M University, 2005.

[5] Fabrizio Biscotti, Brian Blau, John-David Lovelock, Tuong Huy Nguyen, Jon Erensen, Shalini Verma and Venecia K Liu, *Market Trends: Gaming Ecosystem, 2011*, Garner, 2011.

[6] Dariush Derakhshani, *Introducing Maya 2011*, Wiley Publishing, Inc., Indianapolis, Indiana, 2010.

[7] Albert T. Ferrer, *The Disciplines*, Game Developer – Game Career Guide, 2, 2008, p. 47–49.

[8] Andrew Gahan, *Game Art Complete*, Focal Press, Oxford, UK 2009.

[9] Game Education Special Intrest Group, *IGDA Curriculum Framework*, 2008.

[10] Katherine Isbister, *Better Game Characters by Design*, Morgan Kaufmann Publishers, San Francisco, USA, 2006.

[11] Ara Kermanikian, *Introducing Mudbox*, Wiley Publishing, Indianapolis, Indiana, 2010.

[12] Matthias Labschütz, Katharina Krösl, Mariebeth Aquino, Florian Grashäftl and Stephanie Kohl, *Content Creation for a 3D Game with Maya and Unity 3D*, Institute of Computer Graphics and Algorithms, Vienna University of Technology, 2011.

[13] Francois Dominic Laramee, *Secrets of the Game Business*, Charles River Media, Herndon, USA, 2002.

[14] Bruce R. Maxim, Benjamin Ridgway, *Use of Interdisciplinary Teams in Game Development*, ASEE/IEEE Frontiers in Education Conference, 2007, Milwaukee.

[15] Michel E. Moore, *Basics of Game Desing*, AJ Peters/CRC Press, Taylor & Francis Group, Boca Raton, USA, 2011.

[16] Mayur Patel, *The Production Pipeline*, Webinar, MP4, FLV, <URL: `http://www.clockandflame.com/downloads.html`>, 6.12.2009.

[17] Evan Pipho, *Focus on 3D Models*, Premier Press, Incorporated, Independence, KY, USA, 2002.

[18] Andrew Rollings and Dave Morris, *Game Architecture and Design: A New Edition*, New Riders Publishing, USA, 2003.

[19] Richard Rouse, *Game Design: Theory and Practice*, Second Edition, Wordware, Plano, TX, 2005.

[20] Brandon Sheffield and Ryan Newman, *Tenth Annual Salary Survey*, Game Developer – Game Career Guide, 2, 2011, p. 23–29.

[21] Seth Spaulding II, *Team Leadership in the Game Industry*, Course Technology PTR, Boston, USA, 2009.

[22] Patrick Stacey and Joe Nandhakumar, *Managing Projects in a Games Factory: Temporality and Practices*, Innovative Manufacturing Research Centre, School of Management, University of Bath, UK, 2005.

[23] Bill Stoneham, *How to create Fantasy Art for Computer games*, A&C Black Publishers, London, UK, 2010.

[24] Brad Strong, *Creating Game Art for 3D Engines*, Charles River Media, Boston, Massachusetts, 2008.

[25] Jeffrey A. Okun and Susan Zwerman, *The VES Handbook of Visual Effects*, Focal Press, Oxford, UK 2010.

[26] Anthony Ward, *Game Character Development*, Course Technology PTR, Boston, USA, 2008.