

Developing tools for the video games industry



© Kalisto Entertainment

David Lanier
Plugins and Shaders Developer

© Copyright David Lanier 2004

<http://www.dl3d.com>

Developing tools for video games

Presentation :

- What ? Who ? For who ? When? How?
- Experience
 - Kalisto
- About the graphical production workflow...
- Experience
 - Ubisoft, Tom Clancy's Splinter Cell : Pandora Tomorrow
- Questions <http://www.dl3d.com>

Developing tools for video games

2. Developing which kind of tools ?
3. Who develops ?
4. For who ?
5. When?
6. How ?

Developing tools for video games

1. What is developing tools ?

An answer: invest in development to save time on one of more productions

Examples :

- Development of an exporter plugins from a 3D package to a game engine to convert 3D data
- Realize software / plug-ins to help people save time and simplify the production workflow of a game

Developing tools for video games

1. What ?

Different kind of tools :

- Required (like exporters)
- Optionals but may be very important:
 - Shorten your production time

Developing tools for video games

1. Who develops ?

- Engineers of the project / Freelancer
- R&D engineers that get all requests from all projects of the company
- Artists with scripting experience
- Game / level designers

Developing tools for video games

1. For who ?

- Artists
- Engineers
- Game / level designers

Let's see in details...

<http://www.dl3d.com>

Developing tools for video games

1. For who ?

- Artists (modelers, texturers, animators)

Ideally : provide tools so that artists can focus on their creativity and not on the software they're using.

Examples :

- Automation of repetitives tasks
(ex : exporters callable in script, and launched at night)
- Improve the workflow of an existing software using plug-ins
(3DMax, Maya, Photoshop...)

Developing tools for video games

1. For who?

- Engineers :

Examples :

- Libraries (collision, network, AI..)
- Editors
(wheather editor on Kalsito's GT)

Developing tools for video games

1. For who?

- Game / Level designers

Examples :

- Editors / plug-ins to let people design the levels (ex : HTG pos. Graph)

Developing tools for video games

1. For who ?

Another interest of developing tools :
Let people be autonomous by breaking too much dependencies between artists / engineers /level designers

Example : Menu editors in games, usually used to be hardcoded by engineers

→ Now using Flash or custom software

<http://www.dl3d.com>

Developing tools for video games

1. When ?

Theoretically :

During the preproduction

In practice :

At any time ! 😊

Developing tools for video games

1. How ?

- **Iterative process**

- Specifications from the customers
- Milestones and content definition from the spec. for each deliverable. In partnership with the customer.

Iteration:

- For each milestone, feedback from the customer :
 - New needs
 - Obsolete needs
 - Modifications (due to the fact of using the tool)
- Check the content and date of the next milestone with the customer

<http://www.dl3d.com>

Developing tools for video games

Experience at Kalisto Entertainment

+300 people (250 in Bordeaux)

100 3D artists

(including modelers, animators,
texturers...)

<http://www.dl3d.com>

Developing tools for video games

At first : working for artists → Artists didn't trust an engineer telling he was at their service.

They were asking me if I feel capable of doing their requests...

Discovery of the WALL between artists / engineers

Developing tools for video games

Artist vs engineer worlds

→ Different languages. We were speaking of the same things with a different language which resulted in misunderstanding...

I had to break this WALL ...

Example of misunderstanding :

Artists : Set some Gouraud on a mesh

Engineers : Set vertex color on a mesh

Developing tools for video games

Artist / Engineer worlds

The tools developer has to be the
GLUE between the 2 worlds to unify
them in a project or in the whole
company !

<http://www.dl3d.com>

Developing tools for video games

Artist / engineer worlds

The tools developer should act as a service provider !

If we set an excellent engineer on tools development but if he's not able to communicate with its internal « customers » (e.g artists), he will fail to provide tools that are used...

We need humbleness in tools development to be able to listen to the requests and realize them.

We have to accept that it's the user that tell you how he wants to use the tools even if you have 10 years of exeperience...

<http://www.dl3d.com>

Developing tools for video games

So , at first, I had to find artists that trust me so I could rely on them for an interesting feedback

E.g : I have met artists that just erase the tool as soon as they find a bug or can't them as they wanted...

Developing tools for video games

Often, the « customers » are surprised that their request are considered and realized !!

→ If nobody ask you something after you have released a tool, it is that there is a problem or the tool is perfect ! 😊...

→ Usually, the more the user is statisfied, the more he asks you...

Developing tools for video games

A tool that is not used or with unsatisfied users is useless and a waste of time and money !

To prevent this :

- Train people to your tools, spend time with them and be as available as possible. We must show people we are at their service.
- The tools developers should always consider himself as a service provider even if he works as an employee inside a company for another employee of the same company

Developing tools for video games

Finally :

The artists trusting me had become my commercials inside the projects and give me requests for tools on a regular basis...

Today, these people have been part of my first customers as an independent service provider.

<http://www.dl3d.com>

Developing tools for video games



The graphical production workflow

One definition : set of tasks to produce graphical assets. It is usually a unique workflow per project even in the same company.

© Michel Roger

<http://www.dl3d.com>

Developing tools for video games

The graphical production workflow :

Goals :

- Write the list of all tasks to do
- Be able to communicate to each team member his role in the process
- Be able to synchronize team work (schedule, parallelization...)
- Identify dependencies between tasks (also programming – artists)
- Extract information for setting up process to drive the project
- Clarify the workflow (modifications, improvement, debug...)
- Define all software and tools needed for the production
- Validate the production process

Developing tools for video games

Graphical production workflow

Simplified example :

To have a mesh in my game, I must :

- Create it with a 3D package, exexample 3DSMax (.Max + .TGA for textures)
- Export it in my game engine format (.XML)
- Import it in my game engine
- Save it in an optimized run-time format for my target platform (console)

Developing tools for the video games

Graphical production workflow

Sample questions to ask on a given task :

- Name
- Which tools are used to create it
- Average time to create it
- Who creates it ...

Developing tools for video games

Graphical production workflow

Sample questions about data :

- Data format
- Which are the files extensions ?
- Which are the specificities of the data ?

Developing tools for video games

Graphical production workflow

Sample questions about validating an asset

- How is it validated ?
- Who is (are) responsible for the validation ?
- Content and time of the modifications if not validated ?

Developing tools for video games

Graphical production flow

Samples questions about testing :

- Which criterion is tested ?
- Who is responsible of the test ?
- Content and time of the modifications if test fails ?
- Ways of testing ?

Developing tools for video games

Graphical production flow

Then we can for example wonder :

- Can we simplify this workflow ?
- Can we re-use some parts of a previous workflow (past projects) ?
- Where can we save time on it ?

→ Can lead to custom tools development ?

Developing tools for video games

Conclusion :

Tools development :

- Simplify / improve the workflow
→ shorten production time
(= save \$)

Conclusion :

Tools developer :

- Be at the service of its « customers »
- Develop iteratively with as more feedback as possible
- Communicate on tools and train people to use them
- Be the GLUE between different worlds of video games
- Reduce dependencies between jobs (prog, artists, LD)

<http://www.dl3d.com>

A concrete example :

Experience at Ubisoft
on the game Tom Clancy's Splinter
Cell Pandora Tomorrow Online
(next presentation)

<http://www.dl3d.com>

Thank you for your attention !

Questions ?

David Lanier

Plugins and shaders Developer

+33 (0)6 70 01 97 73

<http://www.dl3d.com>

<http://www.dl3d.com>