

GDC 2007 Tech Art Roundtable Notes:

What do tech artists do?

- Tools programmers
- Act as an interface or liaison between artists and programmers
- Act as support staff for artists having trouble with in-house or vendor tools
- Triage art asset issues and development system bugs (e.g. the game doesn't build on my dev kit)
- Design, create, and maintain the art creation pipeline
- Maintain documentation on the art pipeline
- Study artists' workflows, identify inefficiencies, and create tools to increase productivity
- Work to share technology across projects and/or studios
- Write shaders
- Create R&D artwork to pass through in production pipeline tools
- Validate upgrades to 3rd party software packages and create migration plans
- Train studio artists on new features found in content creation application upgrades
- Beta test upcoming content creation applications so they can understand the impact of the changes on the studio. They can also provide feedback to the vendor on changes/features that would benefit the studio.

How can a Tech Artist identify inefficiencies in the art pipeline?

- Pair up with an artist to work with them and study their habits
- Have an artist work in your focus studies lab for a few days and study what they do.
- Talk to the artists. Are they satisfied? Do they have any ideas to improve the workflow?

How can someone evangelize the idea of Tech Art within their studio?

- Demonstrate the efficiencies a script or tool will provide the studio
- Show quantifiable metrics. Don't say, "This would probably save some time", say "In practice this will reduce the time it takes to perform this task by X%".
- Work with the producer to schedule time for you to create a demonstration tool
- Be dramatic! If you have X amount of time on your schedule to perform a single instance of a task work within that timeframe to create a tool that will let you finish your task by the end of time X. Then, show the tool and explain how actually doing the task was a small percentage of the time and now that task will no longer take X amount of time to complete.

- Go grassroots. Write scripts to help people other than yourself. Get the artists on your side and they will start evangelizing with you.

Should Technical Artists also create game art?

- Not if they are a full time Technical Artist. Their critical path is one of creating tools. Having them try to work in a dual role is a risk.
- Technical Artists should understand the process of making game art though. Creating R&D artwork in the pipeline to validate the tools can create assets for the programmers to test with is a great way to understand the pipeline.
- Non-critical tasks like special F/X or lighting can, depending on the circumstances, be assigned to Technical Art.

How do you schedule a Technical Artist's time?

- During preproduction they should be designing and creating the pipeline.
- The pipeline should be complete, and hopefully as close to 100% functional as possible by the time the game is green lighted.
- After that the TA discipline should be scheduled as something akin to a support group. While there will always be a need to create a new one-off tool or to find a way to increase efficiency for the most part the TAs should support the artists. They should ensure the artists are working at their peak.
- Non-pipeline work like shaders, game asset monitoring tools (e.g. something to check the memory use of a game asset), etc... could and should be scheduled during production.

How early should a Technical Artist get involved in the production of a game?

- They should have sign-off privileges on early design work. The TAs will understand the technical issues surrounding the creation of art assets necessary to fit the design.
- They should work with the graphic engine programmers as early as possible to identify features and the tools that might be needed to support those features.
- Even before there are 3d modelers on the team there should be TAs doing prototype and R&D artwork.