

Postmortem: Monolith's *TRON 2.0*

By Frank Rooke

From the start, it didn't take long for many of us at Monolith to recognize that a *TRON* project was a once-in-a-lifetime opportunity, not simply because we believed the film would lend itself to great gameplay, but also because of the movie's status as a cultural icon. As a high school student at the time of the original theatrical release, I remember it piquing my interest in computers and videogames. Whether at the time I fully realized the film's impact or not, it certainly planted seeds that flourished later in my life. Since the start of the project, I've spoken to many people about *TRON*, and I repeatedly get the same kind of story: "It's why I'm into computers," "It's why I'm into 3D graphics," "It's why I'm into gaming."

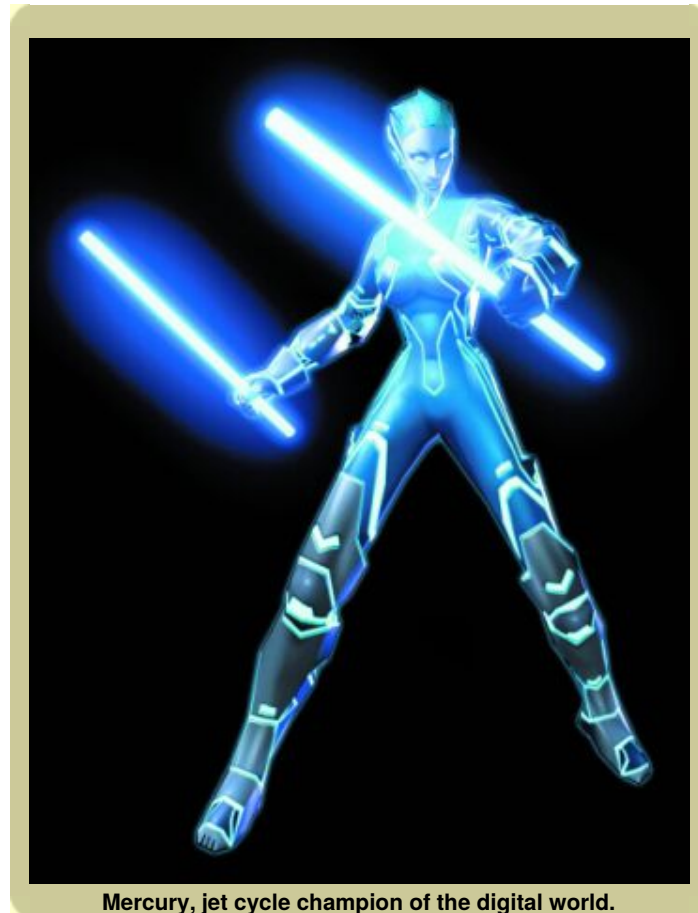


Our intrepid hero, Jet.

When Buena Vista Interactive, the core games publishing label of Buena Vista Games, approached Monolith with the *TRON* project, they were quite up-front about the challenges facing the franchise. While everyone readily agreed it would make a great computer game, generating interest for a title based on a 20-year-old cult classic film that was released ahead of its time might be difficult. Regardless, the project moved forward with great enthusiasm from both Monolith and Buena Vista Interactive. The fact that the game could possibly pave the way for a new *TRON* film and reignite the franchise was very exciting, injecting a unique motivation into the project that Monolith didn't take lightly.

Overall, *TRON 2.0* is a first-person action game that takes place in the digital universe established by the 1982 film *TRON*. It's important to note that the game does not follow the events seen in the film. Instead, it is a spiritual sibling, or something of a sequel. The core premise of a society mirroring our own that exists in the computer remains intact, as does the phenomenon of a human transporting (or digitizing, as we say in the game) into the computer. Beyond that, the *TRON 2.0* universe breaks new ground. Analogies, metaphors, and social consequences reflect how we understand and position computers in our lives today as well as where they may be in the near future. The game tells only one story of a hundred possible stories, making the *TRON* universe much like the Star Wars and Star Trek universes in that

respect. It's this singular quality that makes the *TRON* franchise timeless.



What Went Right

1. Publisher compatibility. It was and continues to be a real pleasure working with Buena Vista Games. We were initially concerned that the constraints of the license would be overwhelming, with minute-level detail examination leading to a potentially watered-down game. However, it was just the opposite. While BVG had great input of their own, they encouraged us to run with our ideas. This freedom afforded us the confidence to pursue a game design without the fear of it changing or being altered in some obtuse fashion down the road.

Another peripheral benefit was the publisher's strong international standing. There are BVG regional offices across the world. Particularly noteworthy are those in Europe. From the onset of the project we had direct contact with the very people involved with press, retailers, and consumers across multiple European regions. From a design point of view, this exposure to non-U.S. markets was enlightening and useful. It's impossible to be all things to all people, but it is good design practice to consider the entire breadth of your target audience. *TRON 2.0* is more accessible and dynamic because of it.





Lastly, BVG granted us access to the archive of Syd Mead, who was involved in the original film. On the art side, we were fortunate to meet with Syd Mead near the beginning of the project, who showed us many of his original *TRON* sketches and paintings. This was a unique opportunity to learn firsthand the design philosophy that created the recognizable elements of the *TRON* world. In a meeting with the *TRON 2.0* artists to pick up where the film left off, we discussed the overall look that is more detailed than the original film, the consistency in the overall aesthetics between the game and the film. Mead also contributed the exclusive design just for the game. Both Monolith and BVG have a rich history of creating iconic elements, and we mention the original light cycle design the next incarnation of the iconic *TRON* light cycle.

Besides Syd Mead, the team at BVG also had a meeting with the original *TRON* designer, John M. Davis, to review progress of the game. Taylor, on one occasion, popped into the Monolith office and provided some very helpful feedback regarding lighting and camera movement. On the acting side, Bruce Boxleitner and Cindy Morgan lent their voices to the game. Most notably, Boxleitner reprised his role as Alan Bradley.

2. Identifying iconic elements from the film. We asked ourselves, what were the core elements that provided *TRON* with its unique identity? Not surprisingly, we immediately isolated the disc and light cycle as iconic elements from the movie and marked them as mandatory features for the game. However, once we started looking past the obvious, we were taken aback by the sheer quantity of other essential *TRON* components. To compound the issue, it became evident that different people - meaning various people on the team, at BVG, in the press, and at *TRON* fan sites - all isolated different elements or events from the movie as true *TRON* moments. What began as a simple checklist became a forum of discussion that never really concluded until the completion of the game.

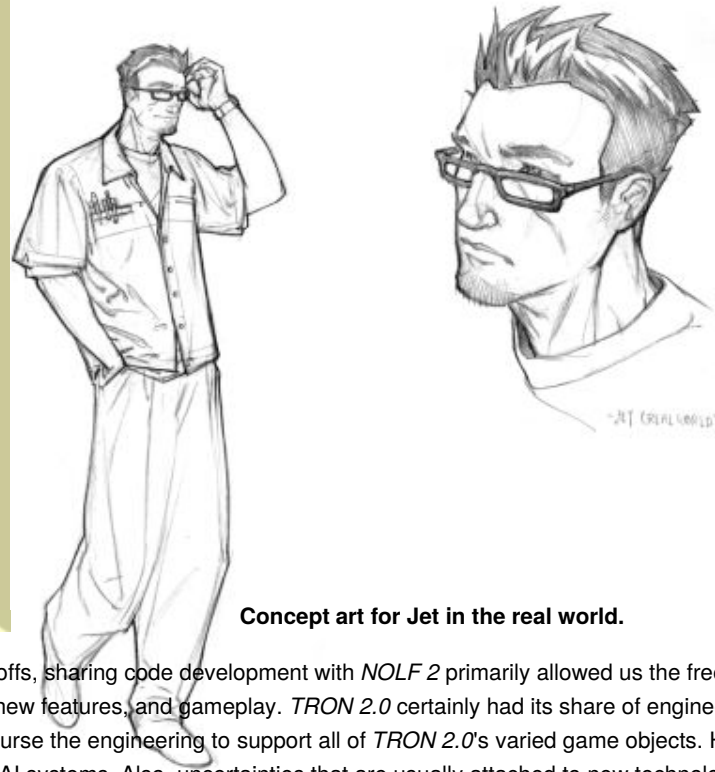
To get a handle on the situation, we started prioritizing signature *TRON* components by how they supported gameplay and to what extent they propagated the *TRON* identity. We then discussed how to mature these concepts to meet the demands of a contemporary game. What we ended up with is a working mix of old and new - recognizable yet fresh. The combat component of the game still revolves around the disc, but it can now be upgraded. Environments retain the glowing, outlined look but with increased vibrancy and complexity. The story is new but resembles the original through the use of playful analogies, techie metaphors, and light-hearted humor - all hallmarks of the original script. And finally, memorable entities such as Bit, Tanks, and Recognizers make appearances but with altered functionality to represent the passage of time.

We avoided simply translating the film directly into a game. It took significant effort to advance the *TRON* universe beyond the safety of the film. Setting out to improve iconic elements is always risky, but as a team we agreed not to take the easy road and shortchange the property's potential by doing the bare minimum. Solely relying on the *TRON* name to sell units was not our strategy.

3. No movie license curse. It's a common belief that movie license-based games are substandard. How many times do we see game reviews with the comment "Game X is just another mediocre game based on a movie." We did not want *TRON 2.0* to be another movie tie-in casualty. Not only would it be bad for Monolith's reputation, but we genuinely didn't want to waste the opportunity. *TRON 2.0* needed to be able to stand on its own as a fun, engaging, and intelligent game, regardless of its lineage.

To help realize this goal, we began work on *TRON 2.0* as we do all our projects by reviewing successes and failures in our previous titles or similar titles so we could learn from past errors. *TRON 2.0* is fully contextual to the *TRON* universe yet iterative relative to past Monolith efforts. With solid game design fundamentals learned from past projects, we were left free to explore unique game mechanics, storytelling devices, and technical enhancements that pushed *TRON 2.0* into new territory.

4. Sharing code. The *TRON 2.0* team found itself in a unique position. The *No One Lives Forever 2 (NOLF 2)* team was roughly eight months ahead of the *TRON 2.0* schedule. They carried most of the burden of developing Jupiter, the next-generation game systems and tools needed to make *NOLF 2* a cutting-edge game. *TRON 2.0* was slated to closely follow *NOLF 2* and directly use the Jupiter engine.



Concept art for Jet in the real world.

Although there were trade-offs, sharing code development with *NOLF 2* primarily allowed us the freedom to focus a greater amount of our resources toward content, new features, and gameplay. *TRON 2.0* certainly had its share of engineering hurdles, such as the glow effect, light cycle technology, and of course the engineering to support all of *TRON 2.0*'s varied game objects. However, we didn't have to worry about creating a new renderer or AI systems. Also, uncertainties that are usually attached to new technologies were for the most part already resolved. We simply learned the parameters of the engine and adjusted the scope of our game to fit.

5. Evolved art direction. *TRON 2.0* has received praise for its colorful architecture, glowing streams of energy, and creative level design. Without a doubt, the artists and level designers on the *TRON 2.0* team successfully captured the essence of *TRON*. Not only do the characters and environments look like those found in the movie but in some cases surpass them. The art direction of *TRON 2.0* really stands out as one of the primary attributes of the game, especially with the recent trend toward hyperrealistic military games. *TRON 2.0* is a fresh alternative.

The method the art team used to achieve the look of *TRON 2.0* was grassroots in nature. During preproduction, the artists re-created many of the actual sets from the film to get the feel for *TRON*. From there, they evolved the look to represent how computers changed over the last 20 years. It's interesting to note that keyboards, monitors, and circuit boards have changed little over the years. But *TRON* is not about the literal interpretation of computers; it's about the abstract world inside, the world of programs, data, and energy. It is here where more significant advancements have been made, and translating that into three-dimensional architecture was the greater challenge. Unlike building recognizable architecture, such as a warehouse or a subway, the artists and level designers had to develop the means to communicate through an abstract language to express what a firewall or PDA looked like to a program.

The film also had a distinctive glow about it. Initially, we attempted to build the glow directly into the art by using layers of additive textures. However, that proved to be time consuming and somewhat inconsistent. Plus, it was not a practical solution for characters. Collaboration between Monolith and Nvidia engineers produced a technique that generated a glow effect that was processed in real time by essentially applying a second render pass with a blurred effect. Once we saw this for the first time, it was clear *TRON 2.0* was going to have a very special look. The glow effect immediately became an item of note when discussing the game with the press.

What Went Wrong

1. Short on initial resources. *TRON 2.0* got off to a pretty good start. A lot of enthusiasm, ideas, and excitement flowed through the office. However, the majority of the team was still busy wrapping up other projects, and only a core team of about four or five actually began preproduction work on *TRON 2.0*. This is a pretty common scenario for most developers. The pacing of production usually has a small team jump-starting the next project by writing documentation, doing concept art, prototyping, and so on. Unfortunately, we failed to recognize a few issues that caused ripples later on in the production. These are lessons that we definitely plan to implement into future projects.

The most obvious was the lack of ramp-up time - not just in skills needed to use the new tools, but also in the mentality and spirit of the project. Past Monolith projects have been founded in more easily understood worlds. *TRON 2.0* was different. It actually took some time and effort just to wrap our brains around what was *TRON 2.0*'s reality. When production officially started and the entire team was in place, there were quite a few months where output was minimal.

Also, the team's understanding of the game design was affected. Scads of

design documents were written up in an effort to explain the concept of the game to the team. This worked to some degree, but was not ideal. Simply telling the team what to do not only failed to tap into the wealth of talent they had to offer, but also created a division in their perception of investment. Being involved during the inception of an idea, or actually being the one who spearheads the idea itself, fully invests a person into that idea. It's key to have the entire team feel personally attached to the project and not just see it as a series of tasks to be completed.

We did have team meetings throughout the preproduction period, and it's not as if the design was written in a vacuum, but communication could have been better. The ownership of the design should have been more team oriented from the beginning rather than something they had to grow to accept over time.

2. Levels unplayable until later in project. It wasn't until very late in the schedule that the team was able to experience a full, playable level. Fortunately, the gameplay came together nicely and in some cases exceeded our expectations, but it would have been a disastrous situation had it not. Why it took so long for the game to hit its stride may again be attributed to our underutilized preproduction period.

As I mentioned in the previous section, it took the team a considerable amount of time to "find" *TRON* - to collectively drive toward a unified vision. We had the necessary talent, and there was no shortage of ideas or inspiration, but the caliber of game we wanted to make was elusive at the beginning. In addition, *TRON 2.0* pre-production did not include a fleshed-out prototype. Light cycle racing, disc combat, and the functionality of the subroutine system all loomed as unknowns for too long. To benefit fully from a good idea is to learn how to exploit it. Having key components of our game remain unknown generated trepidation when it came to linking time and resources. When the game finally did come together and we could see firsthand what worked well, we had very little time to build and expand confidently on those successful game mechanics.

3. Linking projects. Under "What Went Right," I mentioned that *TRON 2.0* benefited greatly from the fact that we used the Jupiter system, which was developed and tested during the production of *NOLF 2*. This was true, but there were a few hurdles that also had negative impact on production.

Although the Jupiter systems were extremely flexible, and even more so now, at the time they were tailored for a more realistic game like *NOLF 2*. To alter or add functionality for the *TRON 2.0* project meant lots of tweaking. In addition, the team preferred to limit changes to game code, rather than rewriting core engine components unless absolutely necessary.

4. Loose review process. During our internal postproject evaluation, one topic that consistently cropped up was the inadequacy of our internal review process. Multiple elements drive the production of any project: an individual's task schedule, the milestone schedule with the publisher, the E3 demo and press schedule, and finally the team's internal progress evaluation. In a lot of ways the team's own evaluations can be the most critical, and it is here that *TRON 2.0* suffered some bumpy times.

In the beginning, we had in place a rather loose review system. It served the purpose of keeping the team on schedule to meet the necessary publisher milestones. What it failed to do was verify the project's overall quality and playability in a timely fashion. It was a strange sensation knowing that we were progressing according to the schedule but worrying deep down that we may not be hitting the mark. We eventually started to evaluate closely the work done by each team member at regular intervals, creating priority lists of subtasks that we felt were necessary to complete before the scheduled task could be crossed off. Toward the final third of the project, we had in place a very comprehensive review process that included weekly and sometimes daily reviews, a cross feedback system from all disciplines on the team, and an aggressive commitment not to falter, regardless of everyone's tight schedules.

Establishing a strong internal review system was so beneficial during the final stretch of *TRON 2.0* that it is now one of our primary management goals for our next project.

5. Commercial 3D software woes. Level design reached a crossroads during the production of *TRON 2.0* when we began using a commercial 3D package to construct our levels. The increased power and flexibility allowed us to model and texture the unique environments found in the game. However, abandoning our game editor during this phase of construction presented gameplay-related issues later in the project.

In the past, constructing the world in our proprietary game editor allowed us to bounce back and forth easily between building and testing. When working strictly in a commercial package, the testing of work required the designer to jump through a few hoops to test work. While not overly complicated, it did require time, and therefore testing was done less frequently. Consequently, levels had problems regarding scale, flow, polygon counts, and delayed functionality - all things that could have been easily checked and verified in the game editor.



Concept art for the Mercury.



Final art for Mercury.

A great deal of time was lost reworking levels to make them playable. Fortunately, now that our designers are more in tune with the workflow between commercial tools and our game editor, this issue is mostly a thing of the past.

End of Line

One of the more remarkable things about *TRON 2.0* is how an eclectic group of individuals came together and weathered many storms to prevail, not with just a finished product, but one of which we can be proud. The dedication and talent the team displayed excites us looking ahead to our next project.



Initial concept art for Jet, it was eventually abandoned for the more heavily armored blue look.

As far as the game itself in concerned, *TRON 2.0* was truly the opportunity of a lifetime. I was personally unprepared for the amount of enthusiasm *TRON 2.0* garnered from fans and from the press, both in North America and in Europe. We knew all along it was going to be a huge responsibility bringing the *TRON* franchise into the 21st century. Now that all the hard work is behind us, it's a thrill to know that what we've accomplished can potentially be the seeds from which future projects can grow. And as a huge fan of *TRON*, I can't wait to see the next tale told inside the world of computers.

Game Data



TRON 2.0

Publisher: Buena Vista Interactive

Number of full-time developers: 21

Number of part-time developers: 4 - 5 pulled in as needed plus the use of Monolith's internal sound, music, and motion capture facilities and personnel.

Contractors: Cinematic music scoring, motion capture actors, voice actors

Length of development: 2 years

Release Date: August 26, 2003

Target Platform: PC

Development Hardware: Pentium 1.0 - 2.0 GHz machines with 256 - 512MB RAM, GeForce 1 - 4 video cards

Development Software: Lithtech DEdit/ModelEdit, Microsoft Visual Studio (C++), Photoshop, Maya, Editplus 2

Notable Technologies: Lithtech Jupiter Development System

Project Size: 2,400 files, 853,300 lines of code

[Return to the full version of this article](#)

Copyright © 2016 UBM Tech, All rights reserved