

Building Worlds with Third-Party Tools

Valve Software

Case Study

“We developed our own AI system completely from scratch, using the *Quake** engine as a base. You see it in the intelligence behind the creatures and the way they interact with the player. Because we didn’t have to rebuild the game engine underneath the program, we were able to concentrate on some of the problems—such as AI and animation—that you can’t focus on if you’re also building the file system and the basic geometry for your 3D world.”

Jay Stelly

*Senior Software Development Engineer
Valve Software*

intel®

Where do you start if you want to compete

in the multi-million dollar computer games marketplace? Gamers as a group want to get involved in a totally immersive 3D environment that typically requires the very latest computer technology available and the most advanced programming techniques. Creating innovative games that engage the imagination and stimulate the senses can require hundreds of thousands of hours of development time and testing.

One solution to the challenge of releasing technology-expanding games while cutting development time is the use of third-party tools. Intel works closely with many third-party tools and technology developers, helping them optimize their products and to make them more interoperable. Valve Software capitalized on the benefits of interoperable third-party tools with their award-winning game *Half-Life**, accomplishing programming advances that otherwise would have been impossible for their small staff. In this case study, learn how Valve combined their own expertise and resources with the expertise of others to expand the depth and intensity of gameplay in *Half-Life*. Their success in this project has led to even more reliance on third-party tools in *Team Fortress* 2*, currently under development.

The Road to Success is Paved with Pre-Built Components

PC game development can be brutally competitive. Each successive generation of games builds on the accomplishments of the past. Developers at the front of the pack face many challenges:

- Insatiable demands of serious, devoted gamers

- Evolving hardware architectures
- Increasingly wide gap between performance PCs and basic computers
- Creative and technological challenges of delivering a quality game to market

Contending with these factors and still getting the necessary programming done requires every possible edge. Pre-built components and third-party tools provide developers the needed edge, allowing them to focus their efforts on features that distinguish their game from the competition.

Half-Life succeeds on a number of levels in this crowded but lucrative market, blending a gradually unfolding story line with non-stop action packaged in some of the most smoothly rendered 3D environments ever released on CD-ROM. Part science-fiction tale, part government conspiracy story, part action thriller, *Half-Life* began collecting accolades even before its official release when a non-sanctioned demo version of the game was circulated on the Internet. Interest continued to build when *Half-Life* was voted Most Promising New Game at the 1997 E3 convention. The runaway success of the title has led to a follow-up episode, still under development, and a number of additional awards. The game

has also spawned Internet team-play components, allowing gamers looking for the ultimate challenge to log on to the Internet, find a *Half-Life* server and compete with others who have become mesmerized by the game's realistic 3D action. The Internet also provides a ready forum and distribution channel for home-grown programmers to produce and release their own *Half-Life* levels, a factor that continues to fuel interest in the game and spread its popularity.

After taking a hard look at the 3D game market, Valve targeted their effort towards the PC platform for two reasons:

- Ready availability of high-power programming tools
- Well-established and expanding base of performance PCs with suitable processing power to handle demanding 3D gameplay

Half-Life was produced using a number of licensed technologies, allowing programmers to focus their attention on storyline, artwork, gaming innovations, and artificial intelligence (AI) development. The core of the product is built around a licensed version of id Software's *Quake** engine. This sturdy and well-tested platform has spawned several different series since its origins in the wildly suc-

successful Doom* release. Freed from the rigors of building and testing a complete 3D gaming engine, the programmers at Valve Software were able to make a number of advances in areas they otherwise would not have had time to explore.

Don't Re-invent the Wheel: Make It Roll Farther

One of the key developers of *Half-Life* was Valve Senior Software Development Engineer Jay Stelly. His solid base of experience ranges from producing titles for the Sony Playstation* and 3DO* to recent work on the 3D game engine underlying Virgin's *Nanotek Warrior**. He also served as lead engineer at Tetragon. From his perspective as a seasoned developer, Jay views the use of licensed components as a practical way to minimize risk when releasing a complex program. The Quake game engine, with its long, successful market history, met all the necessary criteria. "When we licensed that Quake engine," Jay said, "we licensed something that had already gone through many development cycles."

Valve was able to extend and enhance the basic Quake engine in a number of ways, including breaking free of the limitation of using a 256-color palette. Valve integrated truer-to-life color into the game by constructing their own 16-bit color rendering components, a feature that can easily be noticed in the smoothly blended color gradations and greater richness of the scenery.

Other improvements included:

- The development of better tools for scripted animation, which allowed very long animated sequences to be included in the storyline
- Construction of an original skeletal animation system
- Enhancements to the artificial intelligence software, resulting in characters and creatures that respond in convincingly realistic ways

Scripting Long Sequences

More so than many similar titles in this genre, *Half-Life* involves players in the game through an involved storyline that follows an intricate series of twists and

turns. This extra emphasis on plot development is made possible by the modifications to the Quake engine, allowing it to support long scripted animations within a very small footprint.

The story opens with Gordon Freeman, a visiting Research Associate to the Black Mesa Research Facility in New Mexico, entering the facility and slowly moving through a labyrinth of elevators and tunnels to the research area. Donning a hazardous environment suit, Freeman enters the chamber where the experiment is to take place, but an equipment failure causes an explosion and unleashes an army of mutant creatures that run amok and attack the scientists. As Freeman and the others attempt to get back to the surface, they discover that a commando team with unknown allegiances is intent on eliminating any witnesses to the failed experiment. Besides dodging mutant creatures, Freeman must evade well-armed teams of commandos who are trying to seal the facility and suppress the details of the event.

As an example of the extended scripted animation, the long opening sequence in which Freeman travels deeper into the facility goes on for several minutes, following the route that he will have to use when escaping during his trip back to the surface. This sequence skillfully sets the tone for the adventure to follow, as the open train in which Freeman is riding clacks over the rails through echoing tunnels and descends down deep elevator shafts bathed in eerie lighting.

Adding Form and Intelligence to Characters

Other Quake game engine modifications included animation enhancements. "We knew up front that we wanted to have richer characters," Jay explained. "We built a full-skeletal animation system so that we were able to produce much high-

Figure 1. Rich texture maps, elaborate lighting effects, and intricate details distinguish the *Half-Life* world.



er visual quality animation with far less memory than the conventional animation systems. Lengthy scripted animation sequences in *Half-Life* reinforce the story-line as various AI characters interact throughout the game. Our skeletal animation improvements let us fit a lot of animation in a small amount of memory. That enables us to put in hundreds of extra frames just for one little plot sequence—without making our memory footprint too big.”

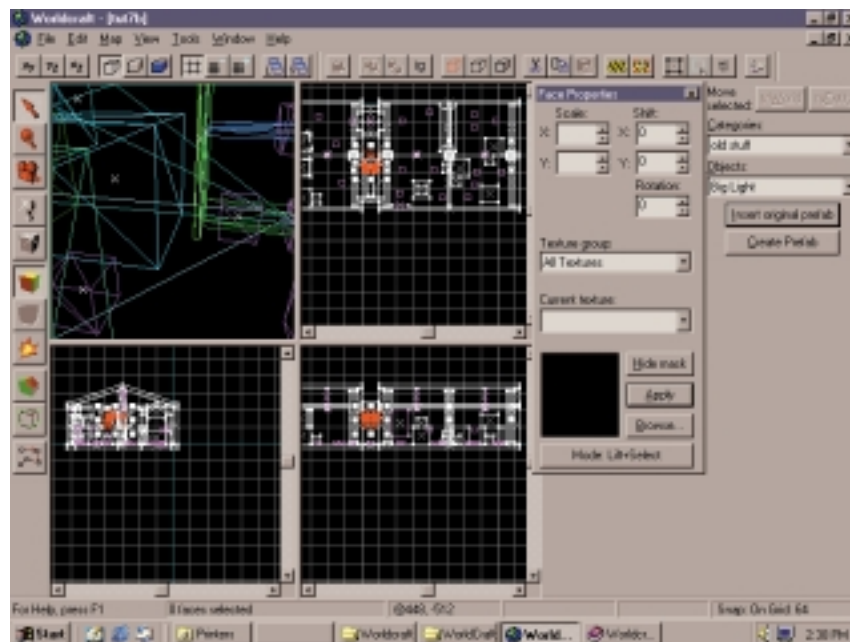
Artificial intelligence was another area where Valve added to the capabilities of the game engine. “We developed our own AI system completely from scratch, using the Quake engine as a base. You see it in the intelligence behind the creatures and the way they interact with the player. Because we didn’t have to rebuild the game engine underneath the program, we were able to concentrate on some of the problems—such as AI and animation—that you can’t focus on if you’re also building the file system and the basic geometry for your 3D world.”

Removing the Barriers from Levels

As a player moves through the game, the distinction between levels is much more subtle than with other action-adventure games. A brief “Loading” message appears onscreen as the code for the new level is loaded into memory, and then the game proceeds. This approach was engineered to make the transitions more fluid and to remove the artificial divisions that levels often impose on a game.

Smoothing the transitions adds some interesting new gameplay possibilities. “This technique made a lot of things possible,” Jay said. “If you’re walking down a hallway with a game character in the standard Quake system, when you change levels, you start on a new level and lose the character following you. In

Figure 2: Worldcraft simplified the process of building and maintaining the 3D world.



our game, we made the transition seamless. It really made the whole world seem more cohesive and held up the suspension of disbelief a little better.”

World-Building 3D Design Tools

Besides just licensing technologies, Valve Software went one step further and purchased one particular product, Worldcraft*, outright. The product’s designer joined the staff at Valve and continues to contribute to their gaming development. Worldcraft, a visual design tool, allows developers very quickly to assemble the basic components of the 3D game environment: buildings, rooms, landscapes, vehicles, and so on. Working in a completely visual design environment saves developers hundreds of hours of design time and leads to richer, more fully realized fantasy worlds.

A complete working version of Worldcraft is included on the *Half-Life* CD-ROM, offering ambitious game programmers the opportunity to try their hand at con-

structing game levels and populating them with entities and scripted events. This capability has led to entire Web sites springing up to support both the development of new game levels in Worldcraft and the distribution of independently developed modules that can be run by the *Half-Life* game engine. The energy generated by this approach is not unlike the pooling of talents in the open source movement. Beginning programmers learn from the veterans. Everyone contributes to building error-free code and useful programs. As with open source coding, the talents of an international body of skilled programmers can achieve remarkable results.

Looking to New Tools for the Future

Flying high with the success of their first title, Valve is using other licensed modules, including new tools from Intel, in the follow-up game, *Team Fortress 2: Brotherhood in Arms*.^{*} “We’re trying to license technology wherever it makes sense,” Jay said. “Our strategy in developing a game is figuring out what we need to build. For us licensing is just as good as building, if it solves the hard problems in a way that doesn’t make bad decisions for us.”

Solving Resolution Problems with MRM

One of the components that Valve is incorporating in their newest game is Multi-resolution Mesh Technology (MRM) from Intel, which solves several problems. “We always need to be aware of the widening difference between basic and performance PCs. There is also a wider performance difference across 3D accelerators than there is across CPUs. How do you design your content so that it can be delivered on all the different 3D accelerators? One way to attack that problem is by dynamically controlling the number of polygons that you put on the screen.”

Character models typically require 5000 or more polygons to display at high resolutions on performance systems. Basic computers, however, with slower processors might be able to handle only 700 polygons. Intel’s MRM technology lets you model a character and then produce a database for that character that supports display at several different levels of resolution. By scaling the resolution to match the system capabilities, performance computers can use the complex, intricate models and basic systems can display a lower-resolution equivalent from the data-

base. MRM also allows characters in the distance to be modeled with fewer polygons, since the details are less important. An appropriate resolution can be selected for varying conditions, a definite advantage when seeking performance enhancements.

MRM provides smooth transitions from high to low resolution on the fly. “Transitions can potentially be abrupt,” Jay said, “as the model gets decimated, moving from a really high-quality, high-polygon model to a low-quality, low-polygon model. We felt good about the choices that Intel had made when developing this technology to ensure smooth transitions.”

Optimizing Code for Processor Efficiency

Valve continues to explore other uses for Intel architecture, including Intel® Pentium® III Single Instruction, Multiple Data (SIMD) techniques. “We’re working closely with Intel on that as well,” Jay said, “to provide multiple versions of various CPU-intensive routines so that if you have a Pentium III processor, it will fully utilize the SIMD performance.”

Valve is also investigating other tools from Intel and other tool developers that offer solutions for specific coding problems.

Licensing technology is an effective way for a small company to gain the development strides that have been achieved by much larger companies. Jay said, “I think as games get more complex, it is only natural that people will be licensing instead of building everything.”

Intel’s Role in Tool Development

Intel continues its commitment to bringing application developers and third-party tool developers together, creating a forum for exchanging ideas, sampling new technologies, sharing design concepts, and connecting with a resource base of useful components. Intel’s relationship with cutting-edge developers enables application solutions for the Intel® Architecture platform, and provides access to information about the interoperable components that will shape software design.

Licensed third-party tools make it possible for well-understood, robust code sequences—even entire functions—to be integrated successfully into products. Game programmers in particular can devote their efforts to showcasing the more exciting aspects of a technology, finding new ways to inspire and delight

their audiences. The problem-solving modules that are available as third-party tools can guide their efforts. Intel's goal is to enable higher levels of interoperability with third-party tool developers and increase success in support of the game developer community. Intel's own libraries and tools are being widely

adopted throughout the industry as a practical, efficient means of leveraging recent processor advances. As these new technologies are integrated by tool providers and developers working in concert, the gamers of the future are in for a stunning and exciting ride.

For Intel Developer Resources, visit <http://developer.intel.com/drg>

About Valve Software

Founded in 1996, Valve develops entertainment software. *Half-Life* is Valve's debut title. Since its release in November, 1998, *Half-Life* has won over 50 game of the year awards from publications around the world, and was dubbed "The best PC game ever" by *PC Gamer* magazine. Valve's upcoming title, *Team Fortress 2: Brotherhood of Arms* won the 1999 E3 Game Critics' Award for Best Action Game and Best Online Game. More information about Valve is available through the company's Web site at www.valvesoftware.com.