

Context

Like the physical architecture of a building or a city, the architecture of a system has to be adapted to the context in which the artifact built using the architecture will reside. In physical architecture, this context includes the historical surroundings of the work, the climate in which it will exist, the ability of the local artisans and the available building materials, and the intended use of the building. For a software architecture, the context includes not only the applications that will use the architecture, but also the programmers who will build within that architecture and the constraints on the systems that will result.

In building the Darkstar architecture, the first thing we* realized is that any architecture for scaling would need to involve multiple machines. It is not clear that even the largest of mainframes could scale to meet the demands of some of today's online games (*World of Warcraft*, for example, is reported to have five million current subscribers, with hundreds of thousands of them active at any one time). Even if there were a single machine that could handle this load, it would be economically impossible to assume that a game would be so successful that it would require such a hardware investment at the beginning. This kind of application needs to be able to start small and then increase capacity as the user base increases, and then decrease capacity as interest in the game wanes. This maps well to a distributed system, where (reasonably small) machines can be added as demand increases and taken away when demand decreases. Thus we knew at the beginning that the overall architecture would need to be a distributed system.

We also knew that the system would need to exploit the current trends in chip architectures. MMOs and (to a lesser extent) virtual worlds have historically exploited Moore's law for scaling. As a processor doubles in speed, the world that can be created doubles in complexity, richness, and interactivity. No other area of computing has exploited the benefits of increased processor speed in quite the way the game world has. Personal computers designed for games are always pushing the limits of CPU speed, memory, and graphics capabilities. Game consoles push these limits even more aggressively, containing graphics systems far beyond those found in high-end workstations and building the entire machine around the specialized needs of the game player.

The recent change in chip evolution, from the constant increase in clock speeds to the construction of multicore processors, has changed the dynamic of what can be done in games. Rather than doing one thing faster, new chips are being designed to do multiple things at the same time. The introduction of concurrent execution at the chip level will give better total performance if the tasks being run by the chip can in fact be executed at the same time. Without

* In talking about the development of the Project Darkstar architecture, I will generally refer to what "we" did rather than speak about what "I" did. This is more than the use of the editorial "we." The design of the architecture was very much a collaborative project, started by Jeffrey Kesselman, Seth Proctor, and James Megquier, and put into its current form by Seth, James, Tim Blackman, Ann Wollrath, Jane Loizeaux, and me.