system are mostly independent (except when they aren't) and don't change much state in the world. Given any two participants in such a world, the likelihood that they are interacting at any given time is vanishingly small. But nearly every player will be interacting with someone nearly all the time. The result is a kind of system that is embarrassingly parallel but interdependent in a small number of interactions.

Scaling of MMOs and virtual worlds is further complicated by the culture that has grown up around these systems. Both MMOs and virtual worlds trace their descent from the production of video games. This is a design culture that grew up in the PC and console game tradition, a tradition in which the programmer could assume that the game ran on a standalone machine or game console. In such an environment, all of the resources of the machine are at the command of the game program, and problems with the program are confined to the single user playing the game (and, in fact, bugs or odd behavior could often be taken as part of the logic of the game itself).

These games, and the companies that write, produce, and enhance them, are part of the entertainment industry. Teams writing a game are led by a producer, and there are scripts and back stories. The goal of a game is to be immersive, persuasive, and most of all, fun. Reliability is nice, but hardly required. Extensibility is a property of the game, allowing new plot lines and themes to be released as upgrades to the game, rather than a property of the code that allows the code to be used in new and different ways.

The rise of online games and virtual worlds brings this culture into an environment where the requirements are much more like those that are faced by the enterprise developer. With multiple players interacting on a server over the network, the crash of a server brought about by the unexpected actions of a player will affect many other players. As these worlds develop economies (some of which interact with the economy of the real world), the stability and consistency of the online world becomes more than just a game. And as the number of players or inhabitants in these worlds reaches the millions, the ability to scale becomes a primary requirement of any architecture.

Project Darkstar (referred to in the rest of this chapter as simply Darkstar) is a response to these changing needs of the builders of games and virtual worlds. The project, undertaken by a research group inside of Sun Microsystems Laboratories, is an ongoing exploration in the architecture of scale. What makes the project particularly interesting is that it is targeted to the MMO and virtual-world builder, a group of programmers who have very different needs from those that we (as system designers) have been used to. The resulting architecture has much that seems familiar until you look at it closely, at which point you can see why it differs from what your experience told you it must be. The result is an architecture with its own sort of beauty, and an object lesson in how different requirements can change the way you have to think about building a system.