We had a few kinks in the production process, but those were corrected very quickly. Thanks to the resilience we built into the loader and render farm, the printing facility has been able to scale up to handle the volume from many more studios than originally expected, while also enjoying higher production quality.

Conclusion

I could spend much more time and space with fond descriptions of every class, interaction, or design decision, with the devotion of a new parent describing his infant's every burp and wobble. Instead, this chapter condenses a year's worth of effort, exploration, blood, and sweat. It illustrates how the structure and dynamics of the Creation Center architecture emerged from fundamental forces about the business and its context. By keeping concerns well separated and guiding the incremental design and development, Creation Center balanced those forces in a pleasing way.

References

Buschmann, Frank, Kevlin Henney, and Douglas C. Schmidt. 2007. *Pattern-Oriented Software Architecture: A Pattern for Distributed Computing*, vol. 4. Hoboken, NJ: Wiley.

Fowler, Martin. 1996. *Analysis Patterns: Reusable Object Models*. Boston, MA: Addison-Wesley.

Fowler, Martin. "Application facades." *http://martinfowler.com/apsupp/appfacades.pdf*.

Gamma, Erich, et al. 1994. *Design Patterns: Elements of Reusable Object-Oriented Software*. Boston, MA: Addison-Wesley.

Hunt, Andrew, and David Thomas. 1999. *The Pragmatic Programmer*. Boston, MA: Addison-Wesley.

Lea, Doug. 2000. *Concurrent Programming in Java*, Second Edition. Boston, MA: Addison-Wesley.

Martin, Robert C. 2002. *Agile Software Development, Principles, Patterns, and Practices*. Upper Saddle River, NJ: Prentice-Hall.

Download at WoweBook.Com