Deploying a Distributed System: Lessons from the Trenches

Greg Johnson
Concept Shopping, Inc.
Deploying a Distributed System

- Distributed systems frequently involve cross-organization collaboration
  - Start accumulating experience early through collaborative prototypes
  - Early and frequent customer involvement, visibility, course corrections
Deploying a Distributed System

- Will it scale to meet the projected demands on the system?
  - Simulate the environment your system interacts with
    - grocery store shopper behavior
    - physical flight environment
    - sensor outputs
  - Simulate operational loads
Deploying a Distributed System

- Integrate early and often
  - The killer problems are the stealth problems you are unaware of!
  - The only way to eliminate them is experience with operational system
  - Negotiate early non-critical "dress-rehearsal" integration activities where failure is an acceptable learning experience
Deploying a Distributed System

- Include error-handling at all levels: system architecture, design, code, testing
  - When an error occurs, how do the dominoes fall?
  - Insert hooks into the source code that can force error paths during testing
Deploying a Distributed System

- A day without regression tests is like a day without sunshine
  - Nightly cron-based source code checkout, system build, and full regression test run; e-mail results to all.
Deploying a Distributed System

- **Bottom line**
  - Identify project success criteria early
  - Wire feedback based on those criteria into the organization from the start
  - Spend brownie points if necessary to gain experience early in order to flush out stealth problems
  - No nasty surprises, happy customers!