Continuous Integration

Howard Olson

THE CHURCH OF
JESUS CHRIST
OF LATTER-DAY SAINTS

Outline

THE CHURCH OF JESUS CHRIST OF LATTER-DAY SAINTS

- Challenges
- Technology
- Architecture
- Lessons Learned

Challenges

THE CHURCH OF JESUS CHRIST OF LATTER-DAY SAINTS

- Best Practices
- Early Discovery
- Prevent Bad Behavior
- Consistent Environments
- Resources
- Traceability
- Release Notes
- Configuration Inconsistency from Dev to Prod

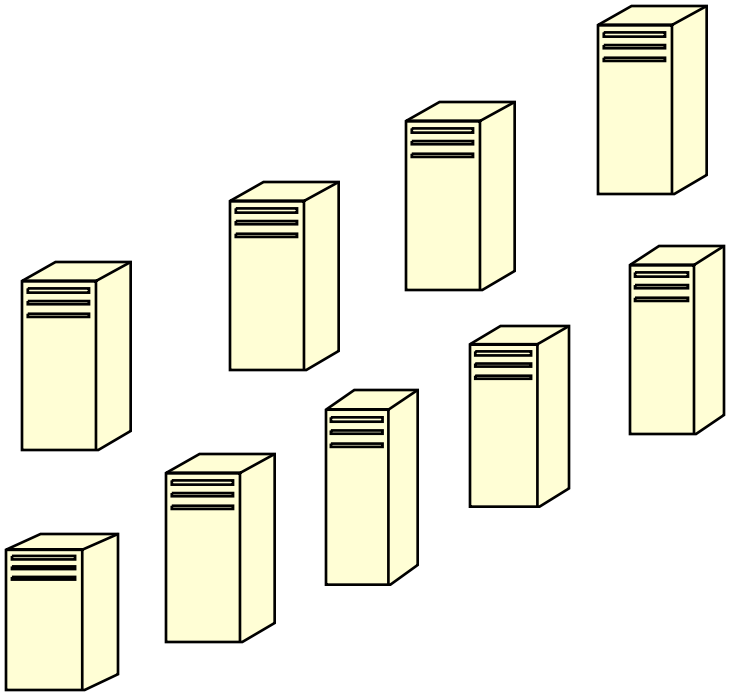
Technology

THE CHURCH OF JESUS CHRIST OF LATTER-DAY SAINTS

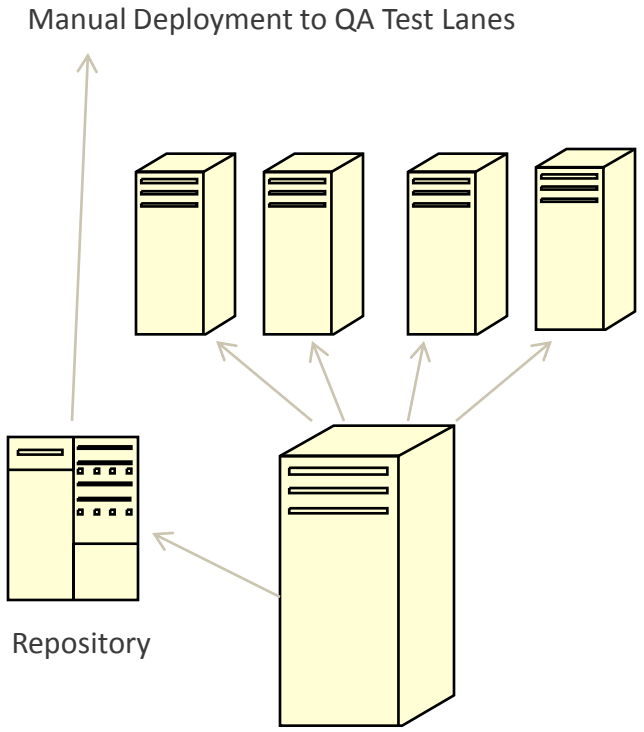
- Linux (SLES) & Windows
- AIX for Web Servers (Linux for dev env)
- Cruise Control
- Started with Ant
- Moved to Maven
- Surefire for junit and testng
- Custom Plugin for WebSphere Deployments
- Clover
- PMD

Architecture – The Beginning

THE CHURCH OF JESUS CHRIST OF LATTER-DAY SAINTS



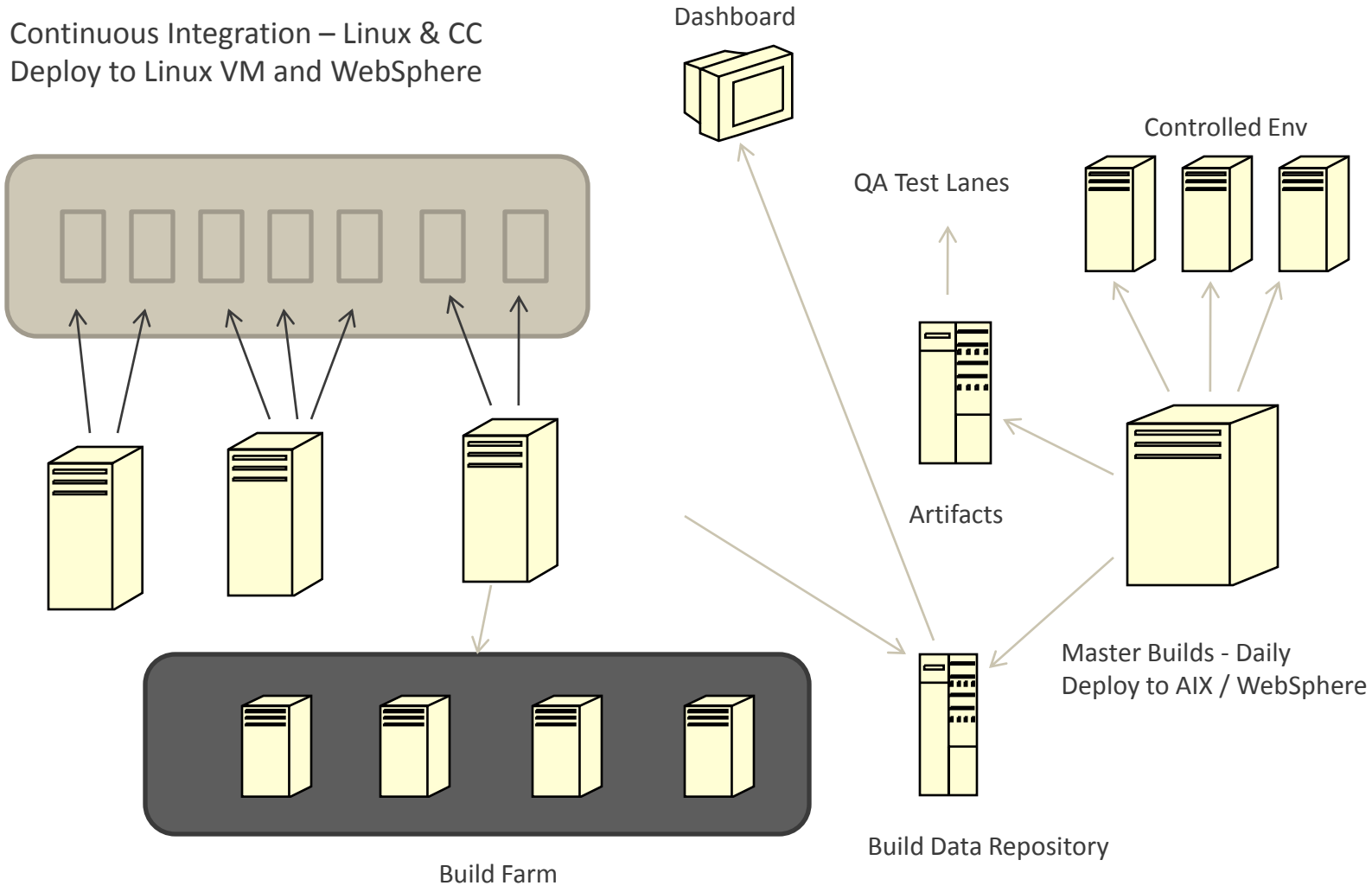
Continuous Integration Env for each Project
Cruisecontrol, Linux
Deploy to local WebSphere environment



Master Build (daily) Environment
Cruisecontrol, Linux
Deploy to controlled env – WebSphere - AIX

Architecture – Today

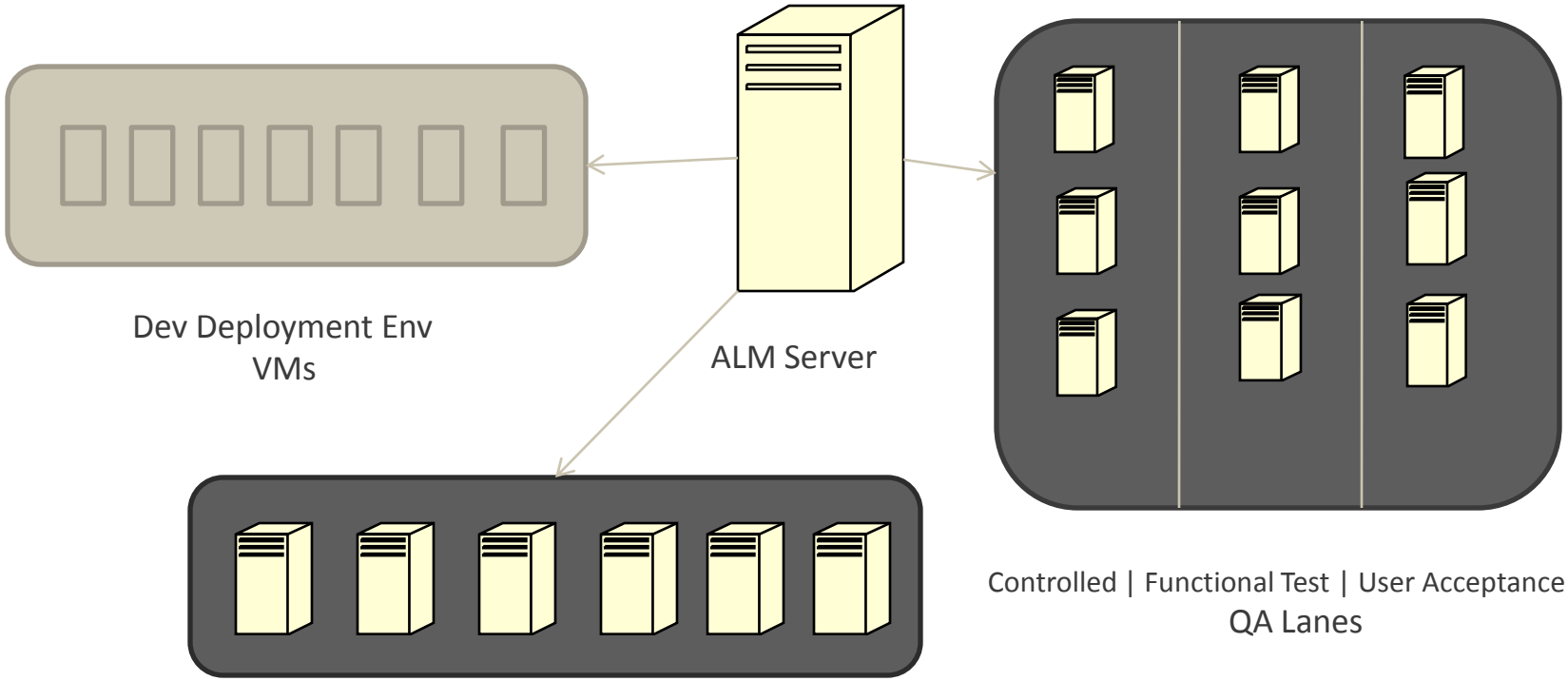
THE CHURCH OF JESUS CHRIST OF LATTER-DAY SAINTS



Architecture – Future

THE CHURCH OF JESUS CHRIST OF LATTER-DAY SAINTS

Application Lifecycle Management



Lessons Learned

THE CHURCH OF JESUS CHRIST OF LATTER-DAY SAINTS

- Report card: A-
- Stay with proven systems and configurations. Stability and consistency is important.
- Worked - Made it easier to use what we offered than to do it yourself.
- Worked - Built features into the build like reporting of metrics and automated deployments- and created a dashboard in common areas so that not using the enterprise solution is immediately and widely apparent.
- Worked - Use imaging and/or VM technologies to make the build box infrastructure a commodity service that can be spun up or rebuilt in minutes and preferably on whatever hardware.
- Not worked - We have outgrown the existing tools. We need to get a more comprehensive deployment and artifact management solution in place (AntHill Pro).
- Not worked - We don't have any sort of common solution across Java and .NET.

Forum Questions

THE CHURCH OF JESUS CHRIST OF LATTER-DAY SAINTS

- What was done to prepare the developers for your build and CI env.
  - We offered training on the use of the tools and did our best to sell the value, not just enforce its use. This training was part of a larger training on the use of our enterprise-standardized frameworks.
- Any particular textbooks you recommend?
  - Pragmatic Project Automation: How to Build, Deploy, and Monitor Java Apps by Mike Clark (Paperback - Aug 2004)
- Were there any specific staffing implications?
  - Strongly recommend creating a build services team, even if it is only one person. This expertise is specialized enough that it isn't worth the cost of spreading it to every team or every developer.
- How do you ensure teams are using your CI env?
  - Reporting from Maven plugins, dashboards that show most recent build time/date, dependencies on the build/packaging operations and automated deployments that developers can't do otherwise.

Forum Questions

THE CHURCH OF JESUS CHRIST OF LATTER-DAY SAINTS

- How have you used consultants or other services?
    - Not specialty consultants, but we have used general contractors to get some of the work done.
- What performance challenges, if any, did you face and how did you resolve them?
    - Too many builds running on a single box. Moving to a build farm with a mix of physical and virtual servers.
    - Tests or functional tests that took too long. Define a set of smoke tests that run during the CI build.