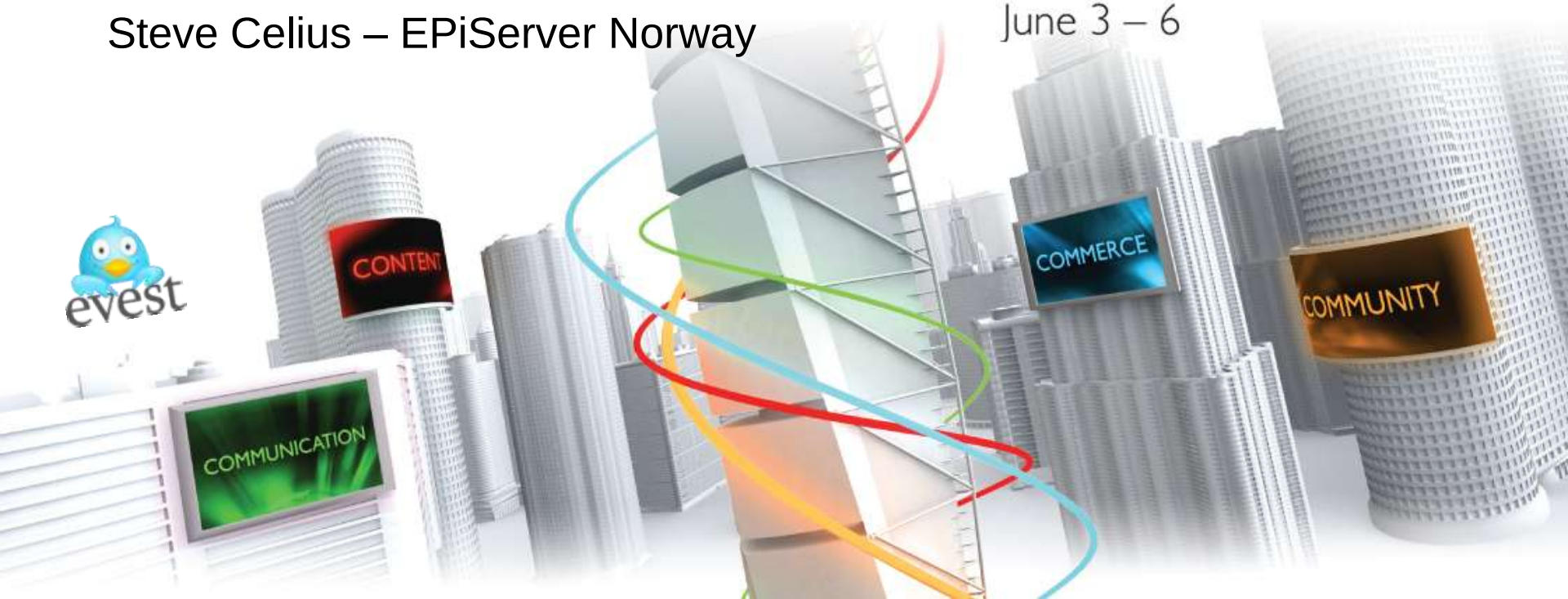
Deployment in EPiServer CMS

Steve Celius – EPiServer Norway

EPiSERVER PARTNER SUMMIT
June 3 – 6



PLAN

EPiSERVER PARTNER SUMMIT

Planning – External Dependencies

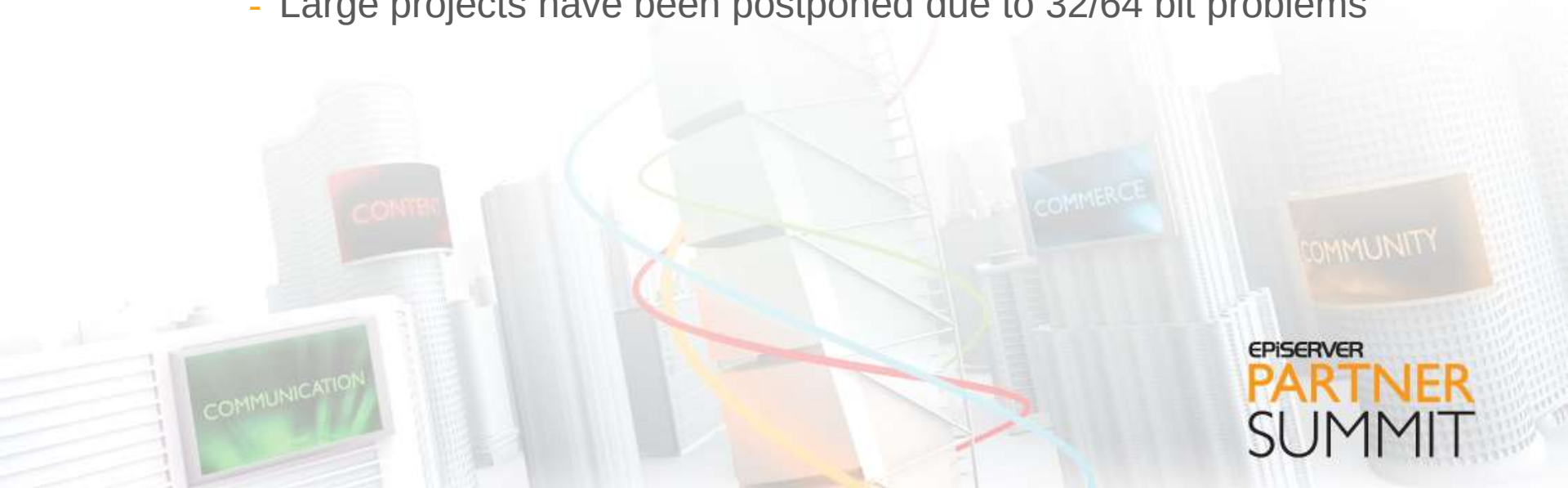
- ”Line Out”
  - RSS Feeds, HR / Vacancies
  - DNS
- SMTP
  - XForms, alerts etc.
  - Most projects needs access to a SMTP server
- Databases (other than EPiServer)
  - Access rights, execute permissions etc.
- Reporting Services or other analysis tools
- Visitor Tracking
  - Google Analytics, SEO Tools, etc.

Planning – Licenses

- EPiServer Licenses for production environment
  - We do not process license requests in weekends
  - You do not want to go live on a demo license (but you could)
- 3rd Party
  - Identify any 3rd party components that you might need a license for

Planning – Operating System

- Test (and develop) on same IIS (major version)
  - Windows 7 for Windows Server 2008 (R1 / R2)
- Develop on 64-bit if you can
  - Windows Server 2008 R2 is 64-bit only
  - Legacy systems / components might make this impossible
  - Large projects have been postponed due to 32/64 bit problems



Installation Basics

- Install EPiServer Deployment Center on the servers
  - Use it to set up sites (install empty site)
  - Copy run-time files to the empty site
  - You will typically move the database using backup/restore
- Configure your services
  - Disable the ones you do not need / use (Log Service?)
  - Multiple Servers? Where does the scheduler run?
- Multiple major versions of EPiServer on same server
  - Test it first – Scheduler could stop working

Virtualization

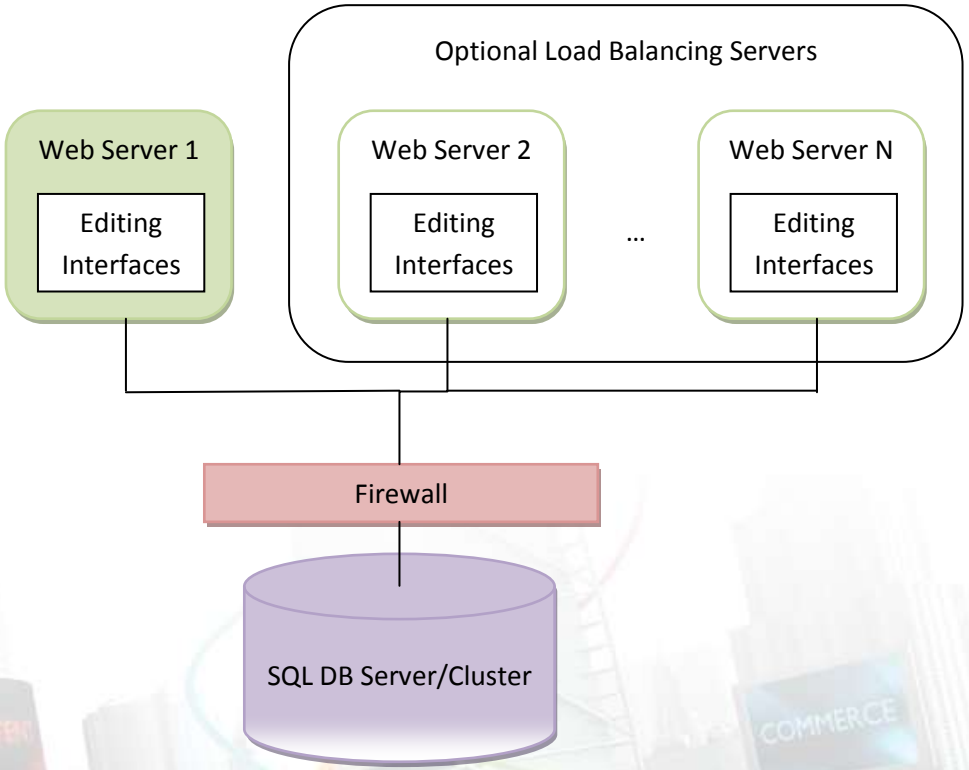
- EPiServer runs fine on virtual servers
- Many customers host like this
- Virtualization makes capacity planning a bit harder
  - Remember: One host = single point of failure
    - even if you have many virtual machines
  - Performance testing is important

Servers server servers...

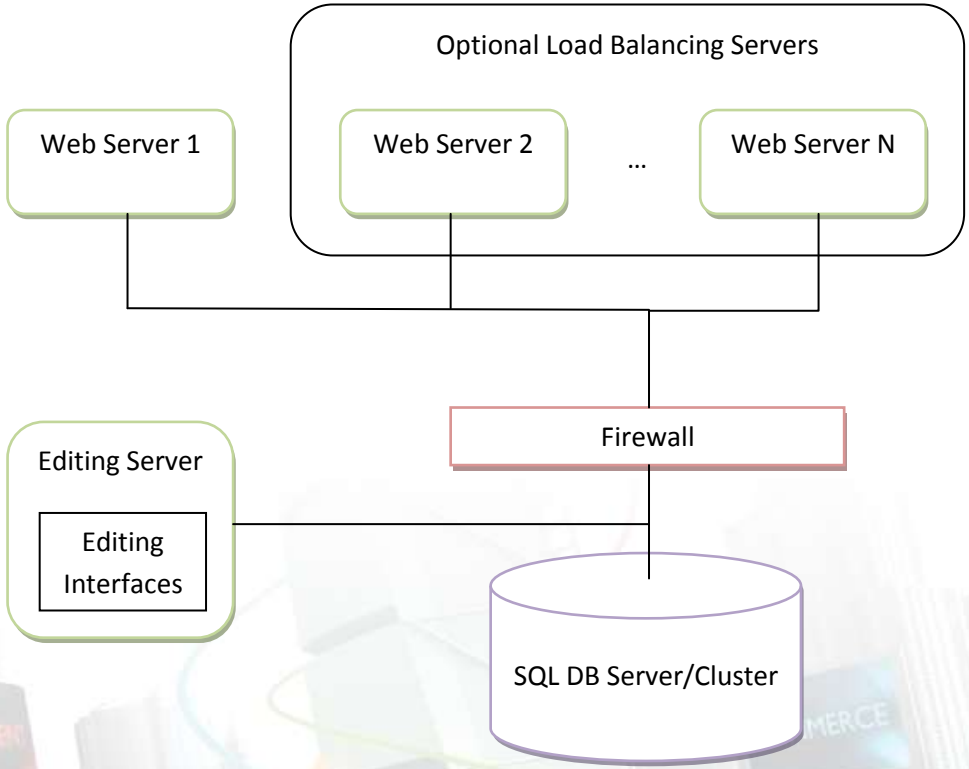
- Important part in planning
- How many servers do you need?
- Web Farms
  - For performance,
  - or availability

EPiSERVER PARTNER SUMMIT

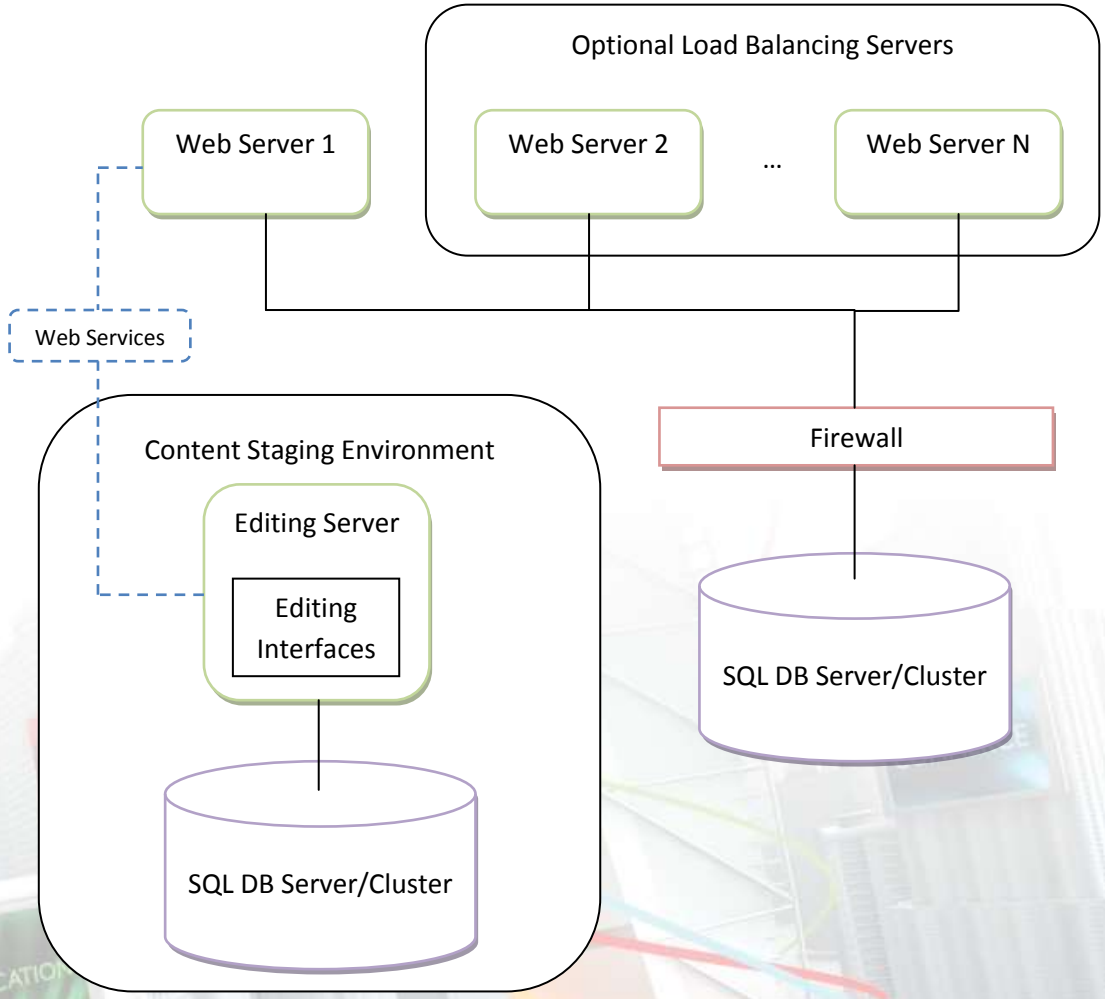
Server Setup – Small Sites



Server Setup – Secure



Server Setup – Content Staging



Single Server vs Web Farms

- High availability / redundancy require more than one server
- Scaling up first could be cheaper than scaling out
- More than one = added complexity
  - Shared VPP storage
  - Cache Invalidation
  - Session Affinity

File Management in Web Farms

- Images and documents are stored on the hard drive
  - Images and documents that editors upload in edit mode
  - In Virtual Path Providers mapped paths
- VPPs must be available on all servers in real-time
  - Or at least close to real-time

File Management – Storage Area Network

- SAN or alternative network storage (NetApp)
- Safe storage (RAID configurations)
- Good performance (most cases)
- Expensive
  - Typically not an option for smaller sites, unless the customer already have a SAN

File Management - Network Share

- The most common option
- No redundancy
- Pick one web server or use database server for files
- A little bit tricky to configure
  - Requires firewall opening for 455 (ms_ds)
  - Run Application Pool as ”other” user
  - Use the same username / passwords on all servers
  - Security on share and disk - requires Modify rights (better on 6)

File Management - File Replication

- Fully redundant – all servers have all files
- File replication is inherently dangerous
- Software for file replication or syncronisation
  - RepliWeb RDS
  - Vice Versa
  - MS DFS (requires AD)
- Synchronising EPiServer VPP folders using the Microsoft Sync Framework
  http://world.episerver.com/Blogs/David-Knipe/Dates/2009/11/Synchronising-EPiServer-VPP-folders-using-the-Microsoft-Sync-Framework/

File Management – Database Filesystem

- All files are stored in the database
- No hazzle, easy to set up
- Require more space for the database
- Caches files on the servers for optimal performance
- Require external search engine
- Free Module on EPiCode:
  www.coderesort.com/p/epicode/wiki/DatabaseFileSystem

EPiSERVER PARTNER SUMMIT

Cache Invalidation

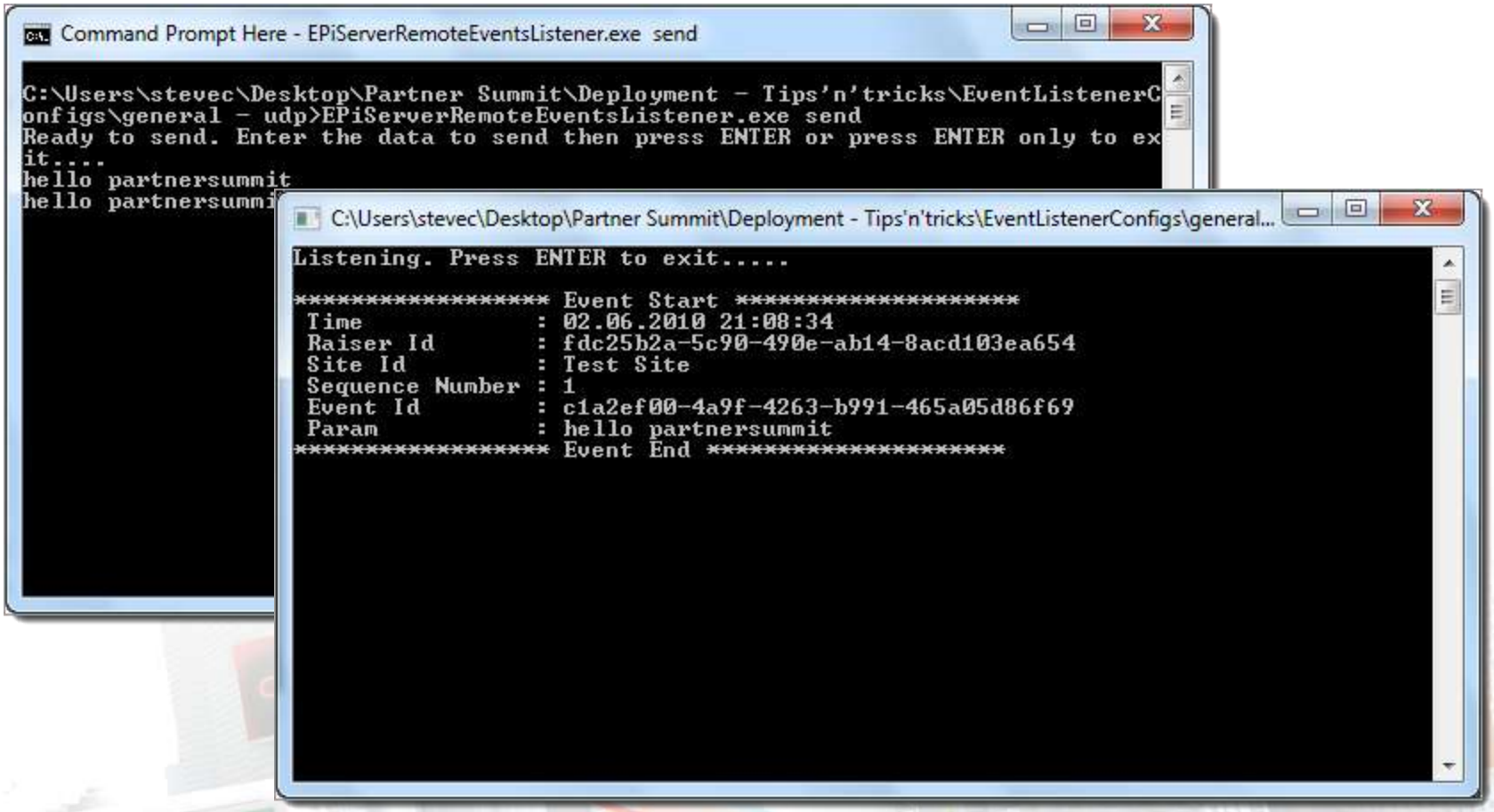
- Cache needs to be cleared on all servers when content is changed
- Use UDP if possible
- Requirements
  - Enable Events
  - Enable Remote Events
  - Add Http Module listener
  - Configure the <system.serviceModel>
- Based on WCF, highly configurable

  http://world.episerver.com/Documentation/Items/Tech-Notes/EPiServer-CMS-5/EPiServer-CMS-5-R2-SP2/Event-Management-System-Specification/

Cache Invalidation – Resources

- Test with the RemoteEventListener tool
  http://world.episerver.com/Documentation/Items/Tech-Notes/EPiServer-CMS-5/EPiServer-CMS-5-R2-SP2/Event-Management-System-Specification/
- Read More:
  - Fredrik Haglund's blog
    blog.fredrikhaglund.se/blog/2009/09/22/episerver-cms-how-to-configure-remote-events-with-many-servers-and-firewalls-between-them/
  - LoadBalancing in 6 steps
    http://labs.episerver.com/en/Blogs/Allan/Dates/112230/11/LoadBalancing-in-6-steps/
  - EPiServer FAQ 228
    http://world.episerver.com/FAQ/Items/Multicast-UDP-not-working/

Using RemoteEventListener



Continuous Integration

“Continuous integration describes a set of software engineering practices that speed up the delivery of software by decreasing integration times.”
en.wikipedia.org/wiki/Continuous_integration

» Some of the recommended practices:

- Use a source control management system
- Automate builds
- Make it easy to see the build results
- Commit often
- Automate deployment

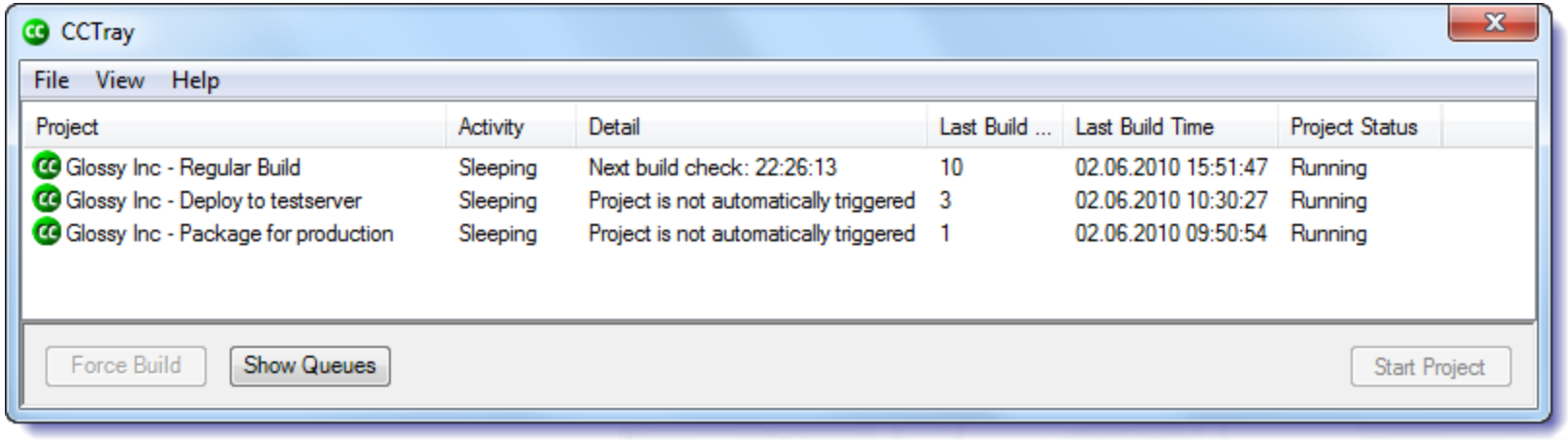
Continuous Integration - Tools

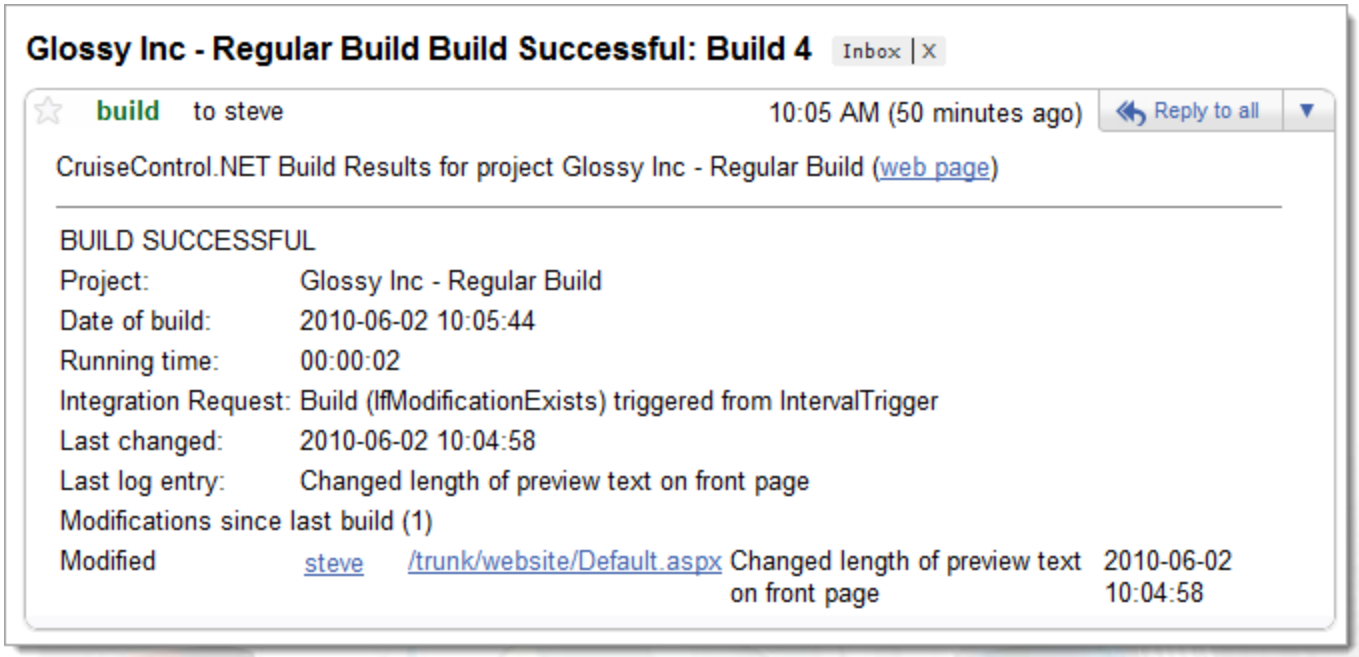
- CruiseControl.NET
  - http://ccnet.thoughtworks.com
- TeamCity
  - http://www.jetbrains.com/teamcity
- Microsoft Team Foundation System

You'll never look back

- Catch build & integration errors early
- Do not compile and package manually
- Add css/js compression etc to your builds
- Everyone on the team should be able to build
  - If not fully automated, pick a deployment responsible
- Invaluable for maintenance
  - Keep the build server around (virtual?)
- Prepare files or do the actual deployment directly from the tool (scripted)
  - Dev/Test could/should be deployed automatically

CruiseControl.NET Tray Application





Configuration – IIS

- IIS 6 vs. 7
  - Oh my
  - Features and stuff
  - Develop on same (or near) production IIS version
  - See tech note: Changes Between IIS6 and IIS7
- Config on IIS 7 is way better than IIS 6
  - Easier deployment
  - IIS Management Console saves to config files
  - Make sure you know about “Feature Delegation”

More Configuration – IIS

- Add server name to http headers in load balanced environments
- Machine Keys can be generated in IIS 7
- Application Pool Settings
  - Recycling on specific times (02:00)
- Turn off Debug (remember to compile for release)
  - 10% better performance
  - Times out during long running requests

Even More Configuration – IIS

- Client caching
  - Set cache timeout for VPPs
  - Use YSlow or similar to test
- Output caching

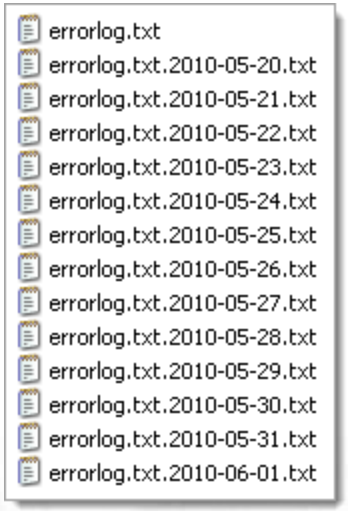
EPiSERVER PARTNER SUMMIT

Configuration Change Management

- Keep configuration under source control
  - Except connectionStrings.config and other security sensitive data
- Split into section files (configSource)
  - Only split sections that differ
- Use diff/merge to move changes to new env.
- Script deployment of config files

Log4net

- Set up Rolling File Appender
  - Beware of buffering
  - File Appender does not buffer
- Use Warn (not Debug)
- Do not log to C:
- Plan to roll logs
  - Do not fill the drive with log files
  - Don’t forget the IIS logs



Security

- Remove IIS features you do not use
  - ”Server Manager” or ”Turn Windows features on or off”
  - http://learn.iis.net/page.aspx/29/installing-iis-7-on-windows-server-2008-or-windows-server-2008-r2/
- Encrypt the connectionStrings.config file
  - "C:\Windows\Microsoft.NET\Framework\v2.0.50727\aspnet_regiis.exe"
    -pef "connectionStrings" "c:\episerver\sites\mysite"
    -prov "DataProtectionConfigurationProvider"
- Remove UI folder on front end server
- IIS Lock Down and other tools
- Stop EPiServer Services not in use

EPiSERVER PARTNER SUMMIT

Security - Continued

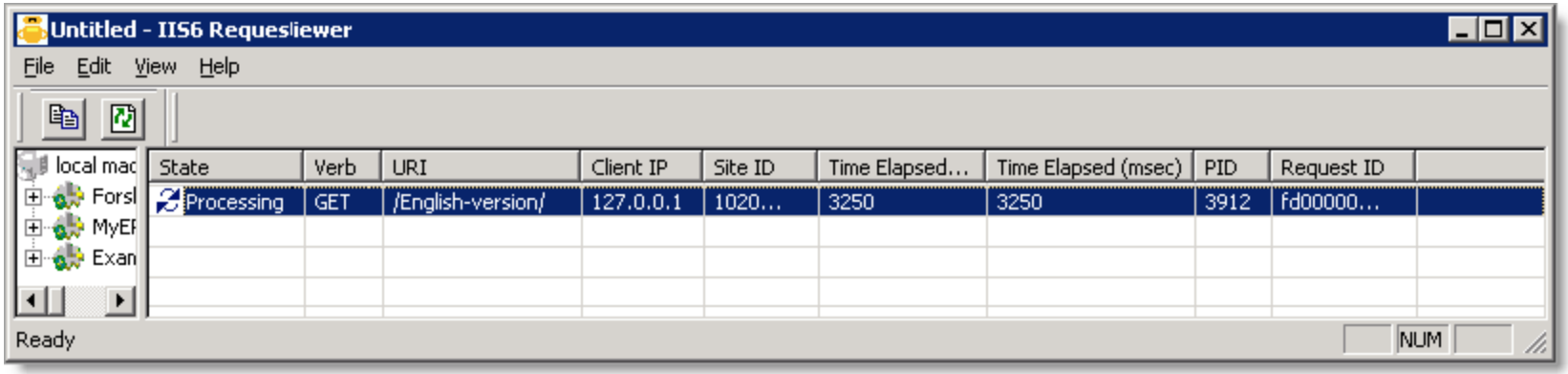
- Remove test users from the database
- Remove test content from the site
- Verify that your folder structure is secured
  - Modules with .aspx files that does not check sec.
  - Editor/admin pages that skips authentication
- Remove “X-Powered-By” header
- Do you need SSL?
  - Forms auth sends username and password in clear text

After Launch

- Having performance problems?
  - Can you use output caching?
- Log to see what happens
- Use Performance Monitor to get a better view
  - Requests / Sec
  - Requests Total
  - Memory Usage (process)
  - CPU Usage (w3 process)
  - Cache utilization
  - Errors Total

Identify bottle necks

- See running IIS Request
  - Win 2003: IIS Trace Diagnostics



  - "C:\Program Files\IIS Resources\TraceDiag\iisreqmon.exe"
- Win 2008: Use IIS Manager
  - Server / Worker Processes / App Pool / Requests

Other Resources

- Checklist for deploying EPiServer sites
  - http://labs.dropit.se/blogs/post/2010/01/10/Checklist-for-deploying-EPiServer-sites.aspx
- Complex EPiServer CMS Projects
  - http://world.episerver.com/Get-Started/Complex-EPiServer-CMS-Projects/

evest