

Optimizing Perforce Infrastructure For Software Build Management

Perforce User Conference

May 2007

Presented by

Slava Imeshev

President & CTO

Viewtier Systems, Inc.

vimeshev@viewtier.com

www.viewtier.com

Outline

- Introduction
- Notion Of A Software Build
- Software Build Management
- Performance Infrastructure Challenges
- Network Optimization
- Architecture Optimization
- Conclusion

Introduction

Viewtier Systems

- Software build management company
- Develops Parabuild, a continuous integration and build management server.
- Perforce users make about 50% of customer base

Software Build

- Process of transforming project code base into a usable application
- Traces its roots to 1952 when the first compiler was devised by Grace Murray Hopper
- Also is used as a communication pattern to define:
 1. Current state of the code base ("build is broken")
 2. Actual results of build process ("build #650 is in production")
 3. Build tools and scripts ("run build at 2:00am")

Software Build Management

Processes And Tools To Deliver

- Build scheduling
 - Integration builds
 - Scheduled builds
 - Manual Builds
- Build execution
 - Sequential
 - Parallel
- Build results notification
- Archive results management
- Security

Integration Builds

- Ensure that new changes don't break the build
- Run on checkin in or a group of checkins
- Run mostly at day time when developers are active
- Impact on Performance
 - Minor if builds are incremental (p4 sync ...@)
 - Significant if builds are required to run in a clean workspace (p4 sync -f ...@)

Scheduled Builds

- Produce QA or UAT builds
- Run according to a schedule
- Used to run nightly
 - To offset load on the system
 - Problem with delayed fixes (no one in the office at 2am)
- Now run at daytime (morning),
 - Can be fixed immediately
- May run on demand, multiple times a day
- Impact on Performance
 - Significant, often require to run in a clean workspace

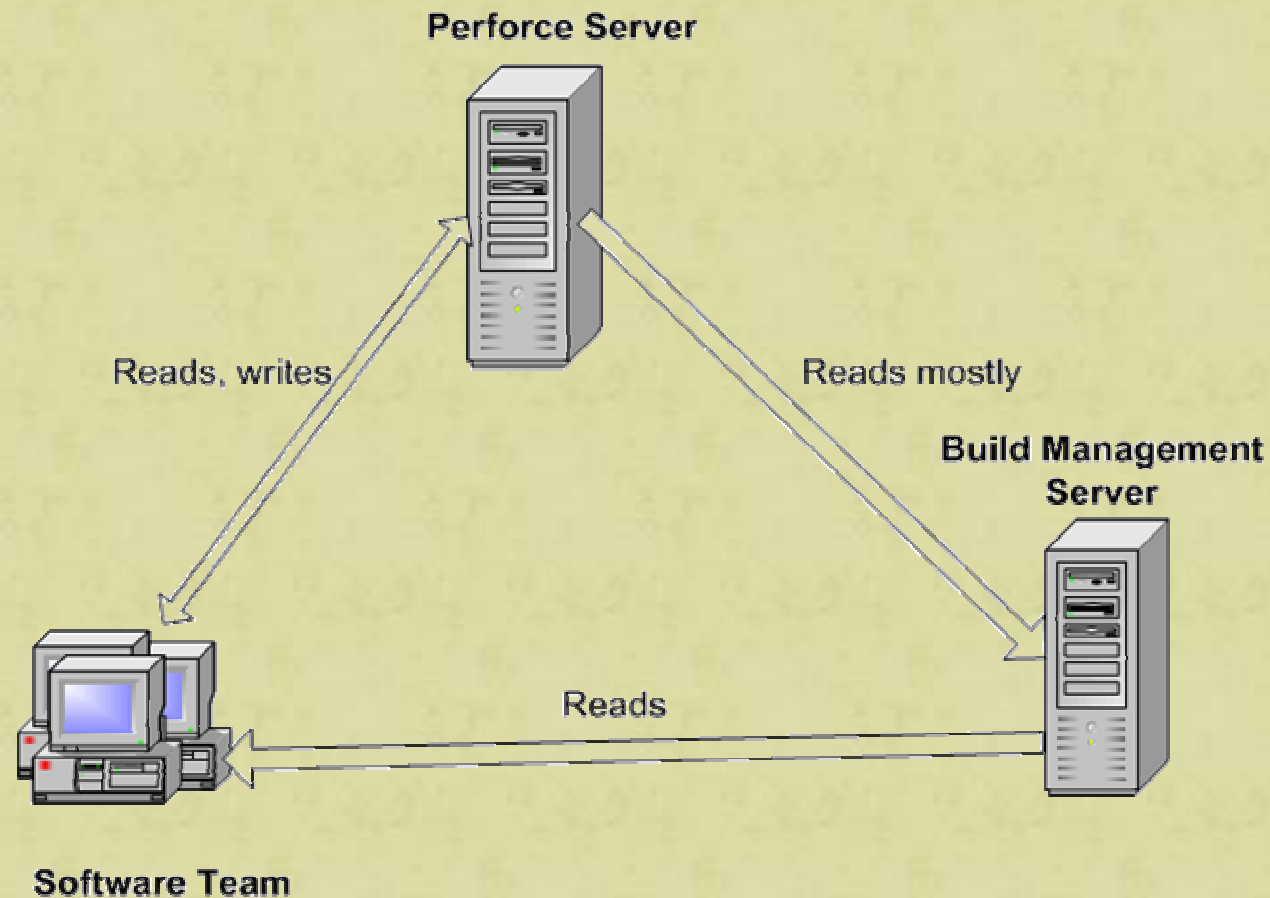
Manual Builds

- Run on demand
- Produce production builds
- Usually run at daytime
- Impact on Performance
 - May be significant, always require to run in a clean workspace

Parallel Builds

- Integration, Scheduled And Manual
- Multiplatform builds
- Build acceleration
- Work separation
 - Build on Linux
 - Image rendering on Mac
- Impact on Perforce
 - Mild if builds are incremental
 - Significant * N if required to run in a clean workspace

Typical Performce Infrastructure



Perforce Infrastructure Challenges

Facts

- Size of codebase grows over time
 - 5 times in 5 years (Example: 300MB -> 1.5GB)
- Number of build configurations grows over time
 - Minimum:
 - 3 build configurations per release branch
 - 30 build configurations in 5 years
- Clean builds often are required for all build types
 - Integration builds
 - Scheduled builds
 - Release builds

Perforce Infrastructure Challenges

Unique challenges

- Build management introduces:
 - Perforce to build management server traffic
 - **High volume bursts** (clean builds)
 - Reads mostly
 - High load on IO subsystem of Perforce (disk, network)
 - Requests for changes (p4 changes –m10)
 - Increased load on Perforce

Symptoms Of Need For Optimization

Watch Performance server for any of

- High level symptoms
 - Check out time grows
 - Time to detect new changes grows
 - Complaints from engineering
- Low level symptoms
 - Average CPU load exceeds 90%
 - Average Network IO exceeds 80% of bandwidth
 - Average Disk IO exceeds 80%

Symptoms Of Need For Optimization

Watch build server for any of

- High level symptoms
 - Build time grows
- Low level symptoms
 - Same as with Perforce
 - Memory consumption (constant swapping)

Optimization Approaches

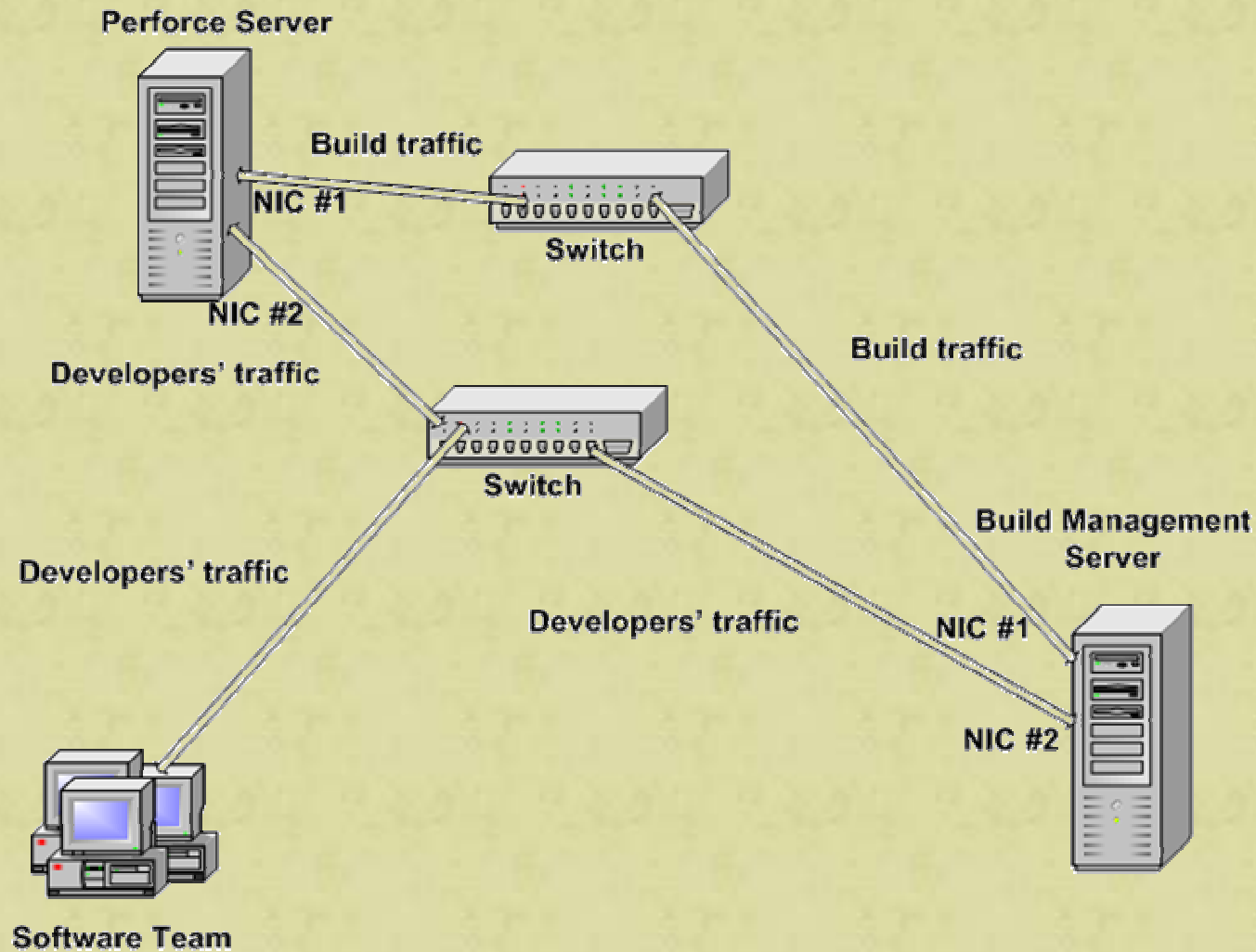
Perforce and high speed build management

- Possible, thanks to Perforce capabilities (fast + p4p)
- Hardware optimization
- Network optimization
- Architecture optimization

Hardware & Software Optimization

- Basic optimization, good for smaller sites
- Get the best hardware money can buy
 - Number of CPUs – min two, or one per case of average concurrent access
 - RAM – the more, the better, watch paging
 - IO subsystem
 - Database, journal on RAID-1 (fast+reliable)
 - Depot on RAID-5 (reliable)
 - Cached RAID + cache battery = \$600
 - NAS maybe not good (1,5GB/s divided by 20 consumers = 75MB/s)
- Scalability is limited to a single box
- Build server software should be able limit number of changes (p4 changes –m1)

Network Optimization



Optimized Network Infrastructure

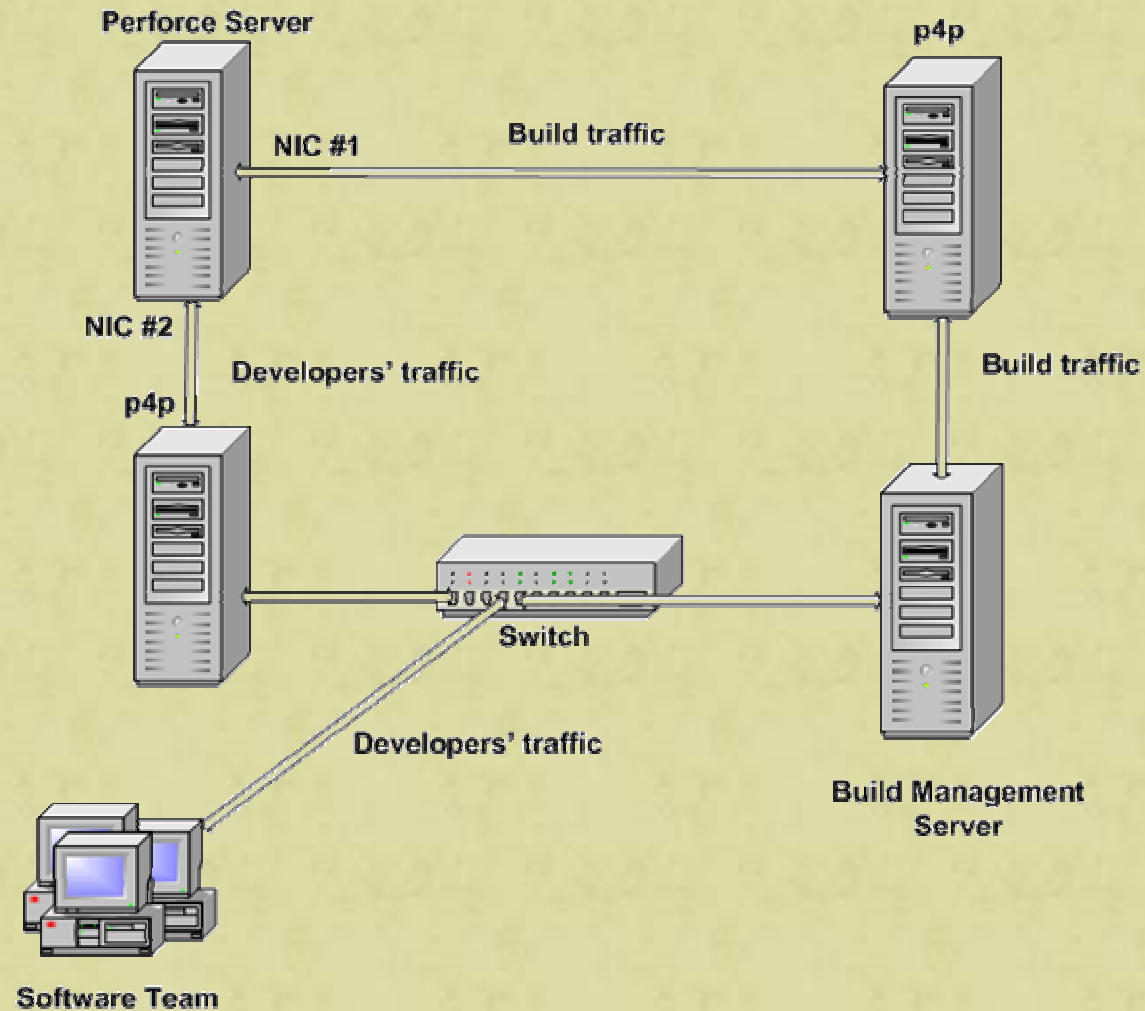
When: Builds generate high volume Perforce traffic

- Requirements
 - Reduce effect on engineering team
 - Reduce locks in Perforce DB
 - Increase speed of p4 sync on build manager
- Solution
 - Isolate build management from engineering
 - Two NICs per server
 - High speed network switches (1-10Gbit/sec)

Optimized Network Infrastructure

- Advantages
 - Easy to implement
 - Immediate relief
 - Low cost
- Problems
 - Overloaded Perforce server
 - May become a bottleneck for large code bases
 - Bus load
 - IO load
 - CPU load

Architecture Optimization



Optimized Architecture

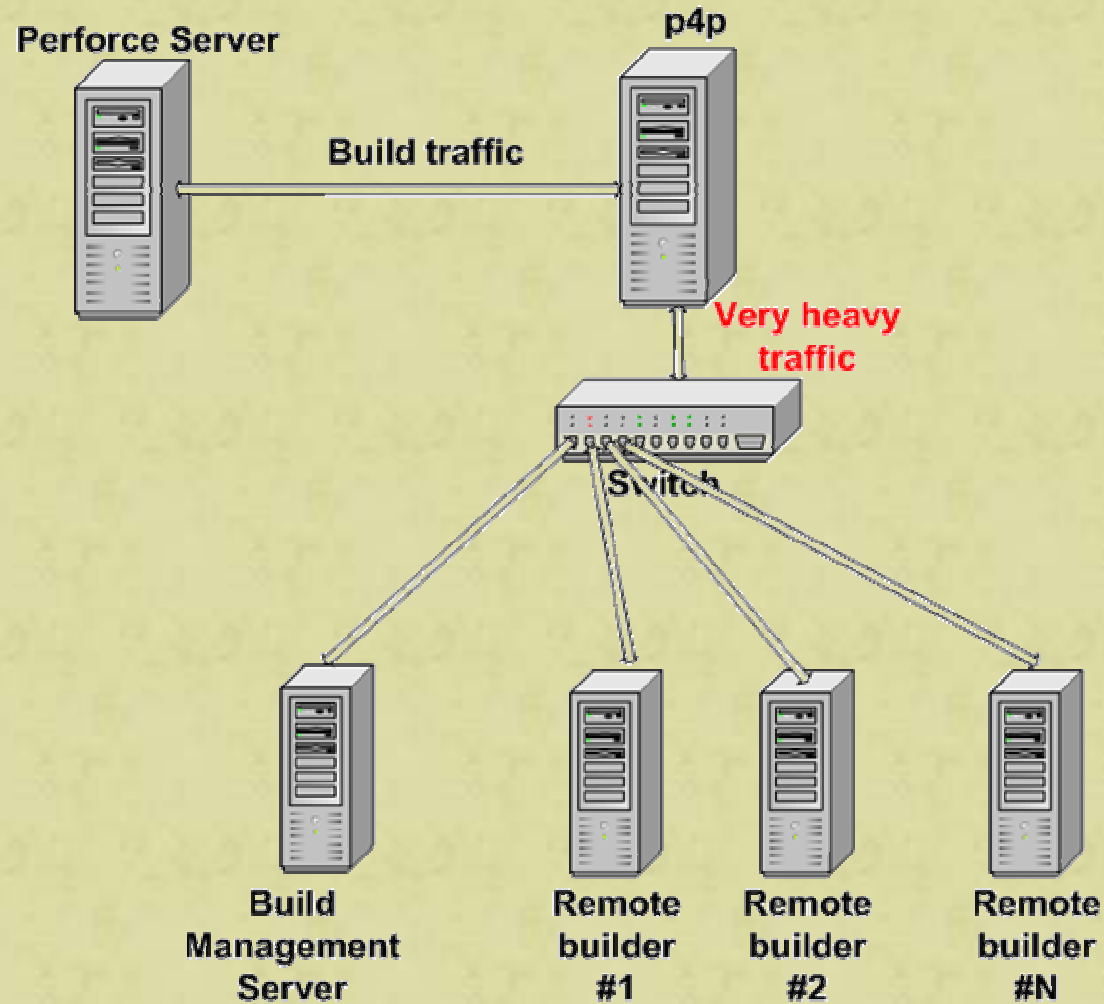
When: High load on Perforce server

- Network optimization may be not enough
- Requirements
 - Reduce load on single Perforce server
 - Avoid traffic bottlenecks
- Solution
 - Isolate build management and engineering behind Perforce proxy servers
 - Two proxy servers

Optimized Architecture

- Advantages
 - Low load on Perforce server
 - Build and developers' traffic don't collide
 - Good enough for developers
- Problems
 - Doesn't scale as number of builds grows
 - Multiplatform builds (N X M platforms)
 - Build acceleration
 - Proxy server servicing build management becomes a bottleneck
 - Requires two big fat boxes

Multiplatform Builds And Build Acceleration



Q & A

Q & A

Build With Pleasure!
Thank you!