

Changing Horses Mid-Stream

Principles of Changing SCM Systems with a Case Study

Stephen Vance, Stellar Advances
Rajul Vora, Google, Inc.

BCS CMSG June 2005

Changing Horses Mid-Stream
© 2005 Stephen Vance & Rajul Vora

1

Overview

- SCM Conversion
- Conversion Challenges
- Case Study
- Conclusions

BCS CMSG June 2005

Changing Horses Mid-Stream
© 2005 Stephen Vance & Rajul Vora

2

SCM Conversion

- Why Convert?
- Why Retain History?
- Conversion Risks
- Conversion Styles

BCS CMSG June 2005

Changing Horses Mid-Stream
© 2005 Stephen Vance & Rajul Vora

3

SCM Conversion

Why Convert?

- Licensing Fees
- Cost of Ownership
- Features
- Usability
- Stability
- Productivity
- Integration with Other Systems
- Auditability

BCS CMSG June 2005

Changing Horses Mid-Stream
© 2005 Stephen Vance & Rajul Vora

4

SCM Conversion

Licensing Fees

- Licensing models for users
- Licensing models for features and add-on products

BCS CMSG June 2005

Changing Horses Mid-Stream
© 2005 Stephen Vance & Rajul Vora

5

SCM Conversion

Cost of Ownership

- Cost of Support Contracts
- Infrastructure Costs
- Administration Costs

BCS CMSG June 2005

Changing Horses Mid-Stream
© 2005 Stephen Vance & Rajul Vora

6

SCM Conversion

Features

- Support for parallel development
- Atomic change
- Remote access
- Platform support
- Interfaces to access the repository
- Security
- Process support

BCS CMSG June 2005

Changing Horses Mid-Stream
© 2005 Stephen Vance & Rajul Vora

7

SCM Conversion

Usability

- Ease of use
 - Simple and intuitive client install (or zero install)
 - Simple and intuitive client software
 - Unobtrusive presence in development process
- Support for agile development methodologies
 - Rapid rate of change
 - Iterative refactoring
 - Conflict resolution

BCS CMSG June 2005

Changing Horses Mid-Stream
© 2005 Stephen Vance & Rajul Vora

8

SCM Conversion

Stability

- Stability of server program
- Stability of client programs
- Stability of platform
- Corruption of repository and disaster recovery procedures
- Metadata maintainability
- Local vs. server storage of metadata

BCS CMSG June 2005

Changing Horses Mid-Stream
© 2005 Stephen Vance & Rajul Vora

9

SCM Conversion

Productivity

- Meeting users' needs
 - Streamlined workflow
 - Parallel development
- Scalability
 - Support for 10 vs. 100 vs. 1000 concurrent users
 - Rapid rate of change
 - Metadata volume
- Geographically distributed development
 - Proxies vs. replication / synchronization

BCS CMSG June 2005

Changing Horses Mid-Stream
© 2005 Stephen Vance & Rajul Vora

10

SCM Conversion

Integration with Other Systems

- IDE Integration
 - Support for development work flow
 - Re-factoring and renaming
 - Network storage vs. local storage performance
- Process Tools
 - Change Tracking
 - Issues / bugs
 - Requirements
 - Change requests
 - Changes
 - Reporting
 - Build
 - Release
 - Distribution

BCS CMSG June 2005

Changing Horses Mid-Stream
© 2005 Stephen Vance & Rajul Vora

11

SCM Conversion

Auditability

- Sarbanes-Oxley in US, Companies Bill in UK, and similar regulations
 - “Reasonable controls to protect the investor”
- Important characteristics
 - Immutability
 - Authentication
 - Linking of identity with operations
 - Reproducibility

BCS CMSG June 2005

Changing Horses Mid-Stream
© 2005 Stephen Vance & Rajul Vora

12

Why Retain History?

- Avoid Vestigial Licensing
- Single Environment
- Reproducibility
- History Browsing

Vestigial Licensing

- Time-bombed license keys
- High cost of ownership
 - Double the licensing costs
- Security
 - Out of sight, out of mind
 - Software bugs, patches unavailable without support contract

Why Retain History?

Single Environment

- Old + New systems vs. one New system
 - End-user training is easier
 - Cross-product code sharing
 - Easier build automation
 - IDE and other tools integration is easier
 - Better ability to support older releases
 - Merge patches into future development

BCS CMSG June 2005

Changing Horses Mid-Stream
© 2005 Stephen Vance & Rajul Vora

15

Why Retain History?

Reproducibility

- Inability to reproduce at milestones means:
 - Inability to fix and release patches
 - Forces maintenance of parallel systems

BCS CMSG June 2005

Changing Horses Mid-Stream
© 2005 Stephen Vance & Rajul Vora

16

Why Retain History?

History Browsing

- Very important for developers
- Crucial for sensitive code (think auditability)

BCS CMSG June 2005

Changing Horses Mid-Stream
© 2005 Stephen Vance & Rajul Vora

17

Conversion Risks

- Feature mismatch
- Inaccuracies
- Lost productivity
- End user rejection of new system
- Direct cost

BCS CMSG June 2005

Changing Horses Mid-Stream
© 2005 Stephen Vance & Rajul Vora

18

Conversion Risks

Feature Mismatch

- Different feature sets
- Mismatch between analogous features in
 - Capabilities
 - Behaviors
 - Representations

BCS CMSG June 2005

Changing Horses Mid-Stream
© 2005 Stephen Vance & Rajul Vora

19

Conversion Risks

Inaccuracies

- Meta-data translation is rarely loss-less
- Feature mismatch requires conversion compromises
- Cost-benefit decisions and prioritization in solving conversion challenges

BCS CMSG June 2005

Changing Horses Mid-Stream
© 2005 Stephen Vance & Rajul Vora

20

Conversion Risks

Lost Productivity

- Down time during conversion
 - Loss of development time
 - Conflict with round-the-clock business drivers
 - Large and active environments have continuous release and QA activities
- Feature mismatch
- Iterative conversions for multiple teams
- Re-training
 - Cost of trainers
 - Time spent in training
 - Acclimatization to new system

BCS CMSG June 2005

Changing Horses Mid-Stream
© 2005 Stephen Vance & Rajul Vora

21

Conversion Risks

End User Rejection of New System

- Change is always hard
- Multiple conversions
- Loss of functionality and ease of use
- Rework of older scripts, processes, integrations

BCS CMSG June 2005

Changing Horses Mid-Stream
© 2005 Stephen Vance & Rajul Vora

22

Conversion Risks

Direct Cost

- New licenses
- New hardware, system software
- Administrator retraining
- Conversion costs
 - Internal development cost
 - External consulting fees

BCS CMSG June 2005

Changing Horses Mid-Stream
© 2005 Stephen Vance & Rajul Vora

23

Conversion Styles

- Full
- Partial Metadata
 - Selective branches and repositories
 - History
 - Branching and merging history
 - Labels
- Milestone snapshots
- Head

BCS CMSG June 2005

Changing Horses Mid-Stream
© 2005 Stephen Vance & Rajul Vora

24

Conversion Challenges

- Scheduling
- Access
- Performance
- Testing and Validation
- Feature, Representation and Behavior

Scheduling

- There is never a good time to pause development
- Minimize conversion window
- Delays can have a ripple effect
 - Conversion is usually on the critical path
 - Integration with other systems typically follows conversion

Conversion Challenges

Access

- Metadata
 - Source
 - Target
- Access Method
 - Direct
 - Indirect
 - APIs
 - Command Line Tools
- Revision Repository
 - Direct
 - APIs
 - Command Line Tools

BCS CMSG June 2005

Changing Horses Mid-Stream
© 2005 Stephen Vance & Rajul Vora

27

Conversion Challenges

Performance

- Source and/or target system too slow
- Each test conversion cycle very long
- Remote location of target / source system
- Algorithmic complexity

BCS CMSG June 2005

Changing Horses Mid-Stream
© 2005 Stephen Vance & Rajul Vora

28

Conversion Challenges

Testing and Validation

- Repository and metadata comparison
- Builds and reproducibility
- Quality of acceptance tests

BCS CMSG June 2005

Changing Horses Mid-Stream
© 2005 Stephen Vance & Rajul Vora

29

Conversion Challenges

Feature, Representation, and Behavior

- Labels
- Attributes
- Triggers
- Metadata Schema
- Users and Access Control
- Branches
- “Sharing”
- Changelists and Changesets
- Directories
- Deleted Revisions

BCS CMSG June 2005

Changing Horses Mid-Stream
© 2005 Stephen Vance & Rajul Vora

30

Case Study: ClearCase to Perforce Conversion at Barra, Inc.

- Project Definition
- Source and Target System Descriptions
- Basic Approach
- Conversion and Test Approach
- Algorithms
- Challenges and Decision Points
- Final Conversion
- Final Verification
- Report Card
- Lessons Learned

BCS CMSG June 2005

Changing Horses Mid-Stream
© 2005 Stephen Vance & Rajul Vora

31

Case Study

Project Definition

- Goals
 - Smooth transition to new system
 - Maximize metadata migration
- Priorities
 - Least disruption to future development
 - Accuracy of migrated data over completeness
- Expectations
 - Uncharted territory
 - Optimistic expectations

BCS CMSG June 2005

Changing Horses Mid-Stream
© 2005 Stephen Vance & Rajul Vora

32

Case Study

Source and Target Systems

- **Source System – ClearCase UCM**
 - Two separate servers / VOBs, one for each product team, geographically distant
 - Typical UCM streams
 - Integration Streams migrated
 - Developer streams not migrated
- **Target System – Perforce**
 - Each product mapped to main and release branches

BCS CMSG June 2005

Changing Horses Mid-Stream
© 2005 Stephen Vance & Rajul Vora

33

Case Study

Basic Approach

- **/main**
 - Converted as one changelist per revision
 - Late-breaking requirement necessitated simplified approach
- **Integration Streams**
 - Map to a codeline in Perforce
 - Activities map to Perforce changelists
 - Each delivery from a development stream maps to a changelist in Perforce
 - Union of activities in developer stream since the last delivery
 - Change description derived from contributing activities

BCS CMSG June 2005

Changing Horses Mid-Stream
© 2005 Stephen Vance & Rajul Vora

34

Case Study

Conversion and Test Approach

- Synthesize Perforce server and merge with production server
 - Perforce journal synthesized from ClearCase
 - Perforce archive synthesized from ClearCase
 - Replay journal file to create test p4 server
- Validate against test p4 server
 - Head revision reproducible
 - File revisions accurate with descriptions derived from traversing delivery -> activities
 - Milestone (label-based) reproducibility

BCS CMSG June 2005

Changing Horses Mid-Stream
© 2005 Stephen Vance & Rajul Vora

35

Case Study

Algorithms

- Mitigate ClearCase performance by creating intermediate files of ClearCase metadata
- Use ClearCase cues to simplify
 - Queries for source and targets of Merge hyperlinks
 - Use activities to create changelists
- Build from /main outward
- Use intermediate schema to support cross-references and index algorithmic searches
- Create Perforce journal files directly to ease creation of arbitrary metadata

BCS CMSG June 2005

Changing Horses Mid-Stream
© 2005 Stephen Vance & Rajul Vora

36

Case Study

Final Conversion

- Each server converted separately
 - Changelist renumbering each time (not very popular)
- First team had single integration stream
- Second team had multiple integration streams (release branches)

BCS CMSG June 2005

Changing Horses Mid-Stream
© 2005 Stephen Vance & Rajul Vora

37

Case Study

Final Verification

- Verification was semi-automated
- Plus manual spot-checks (branching, change descriptions, etc)
- Build reproducibility excellent at head
- Build reproducibility degraded at milestones
 - Typically worse as you go farther back in time

BCS CMSG June 2005

Changing Horses Mid-Stream
© 2005 Stephen Vance & Rajul Vora

38

Case Study

Challenges and Decision Points

- Directory revisions and delivery labels
- Algorithmic vs. manual correction of inaccuracies
- Was history conversion good enough?

BCS CMSG June 2005

Changing Horses Mid-Stream
© 2005 Stephen Vance & Rajul Vora

39

Case Study

Report Card

- Teams transitioned smoothly as head of codelines matched
- Reproducibility at milestones had errors but were fixable in most cases
- File history browsing was excellent
- Vestigial licenses were not required

BCS CMSG June 2005

Changing Horses Mid-Stream
© 2005 Stephen Vance & Rajul Vora

40

Case Study

Lessons Learned

- May have achieved better milestone reproducibility by snapshot method
- In the end, developers cared more about file history browsing than milestone reproducibility but management cared the other way
- Algorithms, challenges tougher than initially anticipated
- ClearCase Base mode may have been tougher depending on usage, UCM mapped nicely to Perforce (mostly)

BCS CMSG June 2005

Changing Horses Mid-Stream
© 2005 Stephen Vance & Rajul Vora

41

Conclusions

- Should you convert?
 - Understand why you want to convert
 - Choose conversion style based on cost-benefit and prioritization
- Advice on conversion
 - Don't underestimate the effort
 - Be prepared for a significant productivity hit
 - Your mileage will vary

BCS CMSG June 2005

Changing Horses Mid-Stream
© 2005 Stephen Vance & Rajul Vora

42