

# Comparison: Perforce and Microsoft Team Foundation Server

---

Perforce  
& Microsoft Team  
Foundation Server

# Overview

This document compares and contrasts Perforce (version 2009.1) with Microsoft Visual Studio 2008 Team Foundation Server (TFS). Included are qualitative aspects such as usability and administration, and quantitative metrics such as the time it takes to branch a codeline.

## Executive Summary

This table summarizes the differences between Perforce and Team Foundation Server.

Characteristic	Perforce	Microsoft TFS
<b>Performance</b>	Perforce outperformed TFS in 18 out of 21 tests. See page 4 for test results..	Performance benchmarks highlight significant differences between TFS and Perforce. See page 4 for test results.
<b>Branching</b>	Perforce supports sparse branching and indirect integration.	No support for advanced features such as sparse branching and indirect integration.
<b>Administration</b>	By using standard file systems, network features, and a single server, the only overhead added by Perforce is associated with backup and other maintenance activities.	Reliance on additional server components such as Microsoft SQL Server, Internet Information Services and Microsoft Office SharePoint Server can lead to higher administrations costs.
<b>Platform support</b>	Available on a wide variety of platforms; includes native support for 64-bit operating systems.	TFS server only available on Windows. No native 64-bit support.
<b>Third-party integration</b>	Integrates with a large variety of application lifecycle management tools.	Tightly integrated with Microsoft Visual Studio Team System.
<b>Support</b>	Provided by Perforce SCM experts and relied upon by over 320,000 users.	Available through Microsoft.

## Performance

To test TFS performance against Perforce, comparative benchmarking tests were constructed to reflect real-life steps that developers take in their day-to-day work. In some cases, individual commands were used. In other cases, scripts orchestrating multiple commands were employed. In all cases,

functionally equivalent sets of commands were executed using the command line utilities.

### Test Environment

The test environment was configured to provide a common development environment that is typical in 2009. The configuration exceeded the minimum recommendations for both TFS and Perforce. The environment consisted of the following:

- **Network:** both server and client on the same segment of a 1Gbps network
- **Server OS:** Windows Server 2008 (Standard) SP1(32-bit)
- **Client OS:** Windows XP Professional SP2 (64-bit)
- **Server Hardware:** Dell PowerEdge 1850 with dual 3.3 GHz Xeon processor X5470, 132GB hard disk and 16GB RAM
- **Client Hardware:** Dell Precision 490 with Intel Xeon 5130 2.0 GHz processor and 2GB RAM

## Test Procedure

Testing began with both source control systems in a newly-installed state. Tests were run on each system alternately. All developer and source import tests were automated and timed using the DOS **time** command to a granularity of .01 second. Software installation was timed externally.

Tests that invoke the file system cache or database cache were run three times, and average times for the second and third iterations were recorded. This approach simulates an environment where many files are accessed repeatedly through builds and source code searches.

A common file structure was used. Each product created, deleted, and modified the same files during testing. File copies were used to simulate file creation, and automated edits were used to simulate file modifications.

When a sequence of commands was required to complete a single test for either system, a DOS batch file was used to execute the test for both systems. For TFS, the recommended **@** switch was used to invoke the command file to reuse a single connection to the TFS server.<sup>1</sup>

No performance-oriented techniques were used to optimize the use of either product, other than those used to remove interactive effects. For TFS, LAN connection settings

in Internet Explorer were configured as recommended on MSDN.<sup>2</sup>

The tests were designed to compare the two products during normal developer usage. For all processing, automated merges were used, as the tests were constructed to avoid file conflicts. The following tasks were tested:

- Server installation
- Client installation
- Importing source from a local file system
- Populating the workspace
- Updating the workspace (with no changes on the server)
- Labeling all files
- Checking out files
- Deleting files on the server
- Undeleting files on the server
- Branching files
- Merging back to the parent branch
- Checking out binary files
- Branching binary files

## Test Data

Test data was a combination of two open source software packages: Apache httpd 2.2.14 and Apache Tomcat 6.0.20. A “small” project contained one copy of each package: 4,480 files totaling 56.3MB. A “large” project contained four copies of each package in separate trees: 33,920 files totaling 417MB. Tests for the large project performed the same operations in each tree. Test data for binary files consisted of 217 files (MPG, EXE, and TAR) totalling 1GB.

## Test Results

Installing TFS was a significantly larger task than installing Perforce, due to the number of applications, services, and service packs involved. The installation footprint for TFS is four times that of Perforce. Perforce out-performed TFS in 18 out of 21 tests. The detailed test results are shown on the next page. All times are in seconds unless otherwise noted.

## Performance Results for Server and Client Installation

Task	Perforce	TFS
Install server	10 mins.	4.5 hrs.
Install client	10 mins.	20 mins.

## Performance Results for Small and Large Projects (all times are in seconds unless otherwise noted)

Task	Small Project		Large Project	
	Perforce	TFS	Perforce	TFS
Import source	9.40	28.15	41.10	56.55
Populate workspace	9.97	24.30	80.70	65.20
Update workspace (no updates)	3ms	9ms	39ms	18.30
Label/baseline all files	11 ms	12.95	36ms	13.70
Check out files	44ms	32.70	5.74	35.30
Delete files on server	2.0	17.15	11.81	30.80
Undelete files on server	27.10	24.80	136.70	62.70
Branch files	9ms	20.75	2.90	35.50
Merge back to parent branch	49ms	10.09	1.34	18.30
Binary files: add files	N/A	N/A	125.05	165.44
Binary files: check out files	N/A	N/A	6ms	17.10
Binary files: branch files	N/A	N/A	8ms	35.50

## Usability

This section compares feature implementation in Team Foundation Server and Perforce.

### Client Specification Model

A user needs to set up a client specification to work with files managed by an SCM system. A user's client specification tells the

SCM system where in the local file system the user's workspace is rooted, through a series of folder mappings.

In TFS, each mapping associates files and folders in the repository with files and folders in the local workstation, creating one-to-one mapping. TFS stores the workspace mappings in a local file on the user's local hard drive.

This approach can pose problems for users if the mappings file contains outdated mappings or an invalid TFS server ID.

Perforce provides greater flexibility; the mapping does not have to be one-to-one. All workspace mapping information is centrally stored in the metadata. This can result in productivity gains, as illustrated in the following scenarios.

### Sparse Branching

Branching a subset of project files as opposed to branching the entire project tree is often referred to as “sparse branching,” or, in some contexts, as creating a “private” or “task” branch. This approach can be beneficial when working with branches that consist of hundreds or thousands of files, but require only a few files to be edited at a time (see Figure 1).

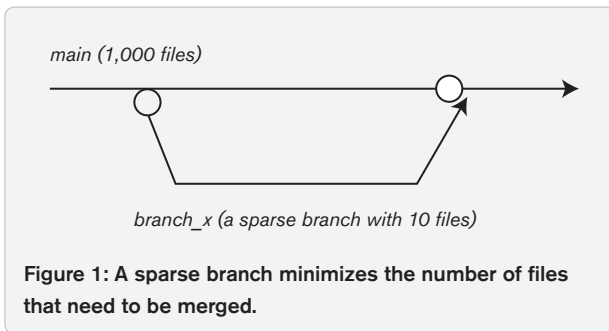


Figure 1: A sparse branch minimizes the number of files that need to be merged.

For example, a release codeline can be branched into a task branch to isolate a risky bug fix. Similarly, a release branch can be branched into a patch branch for an already released version. All the other files (that have not been branched) are referenced from the original branch, so that the client workspace has a complete view of all project files.

TFS does not support sparse branching. A user can individually “cloak” or “uncloak” project folders. However, the same set of files from two different repository folders (for example, **main** and **branch\_x**) cannot

be mapped to the same local folder in a workspace. In TFS, you cannot have two repository paths mapped to one local path on the same computer.

Perforce provides a flexible client specification model that supports the use of “overlays.” For example, a user, “Joe,” might have a client workspace called **joe\_dev** defined as follows:

```
//depot/main/... //joe_dev/main/...  
+//depot/branch_x/... //joe_dev/main/...
```

By branching only the required files from **main** to **branch\_x**, and setting up the client workspace view as above, the workspace **joe\_dev** has access to all 1,000 project files, with the required files from **branch\_x** overlaying the corresponding files from **main**.

### Merging Changes

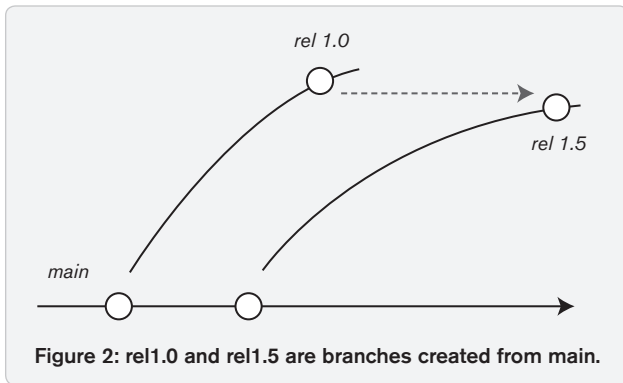
Merging is required when multiple changes made to different copies of a file need to be reconciled. For example, one developer is fixing a bug in a file while another developer is adding new code to the same file. In some cases, the merge can be performed automatically because the changes do not conflict. In other cases, the developer to check in last must decide exactly what the resulting file contains. The merge tool and type of merge (two-way or three-way merge) becomes very important here.

TFS uses a two-pane merge tool. The merge tool shows you files side by side. You can update the result file with the changes you select.

Perforce’s 3-pane merge tool, **P4Merge**, is a powerful and reliable tool for merging that enables users to visualize the differences between file versions. P4Merge provides graphical three-way merging and side-by-side file comparisons, using color coding to simplify the process of resolving conflicts that result from parallel or concurrent development.

## Indirect Integration

Indirect integration enables you to merge changes between two branches that are not directly related. For example, using the mainline branching model, **rel1.0** is a release branch created from **main**, and **rel1.5** is another (perhaps refactored) branch created from **main** (see Figure 2).



Although **rel1.0** and **rel1.5** have a common ancestor, they are not directly related. TFS does not automatically create a relationship between the two sibling branches (branches that share the same parent). TFS can do a “baseless merge” between the two branches (only from the command line, not the GUI).<sup>3</sup> A baseless merge is a two-way merge between **rel1.0** and **rel1.5** branches which assumes they have no past relationship at all. Such a procedure typically results in false “conflicts” that need to be resolved before merging.

Perforce, on the other hand, allows merging between **rel1.0** and **rel1.5**. Perforce is able to determine the common heritage (the **main** branch) and uses it as the basis of the merge operation, resulting in merges free of false conflicts.

## Graphical Tools

Graphical tools help you review source code and branches, diff folders, and compare images. These tools are very useful for

developers, managers, and artists working on development projects.

TFS provides two tools for viewing evolution of source code in a file over a period of time (using the “annotate” feature), and a flat file folder view to compare two folders.

Perforce provides four main **graphical tools** in P4V that enhance user productivity:

- **Time-Lapse View:** provides a graphical view of the complete content history of an individual file
- **Revision Graph:** displays a tree-style representation of the branch history of a specified file, including all file branch points, edits, and merges
- **Folder Diff:** displays two versions of the same folder side by side in expandable tree views
- **Admin Tool:** allows you to visually manage user and group permissions

## Administration

Team Foundation Server is composed of several product components, at a minimum:

- Windows Server 2008
- Internet Information Services 7.0
- SQL Server 2008 in a single-server deployment
- Windows SharePoint Services 3.0 + SP1

It takes several hours to install TFS. The requirement for several product components distributed over multiple servers adds to the training, maintenance, and diagnostic responsibilities of system administrators. Before installing TFS, you must first consider your deployment site: you have the option of a single- or dual-server deployment based on the number of users.

Additional overhead is associated with backups and other maintenance activities required to support the operation of TFS. For example, backing up TFS requires backing

up the data and application tiers. This data includes several SQL Server databases. You must also back up the Report Server's encryption key. For larger projects, the recommended multiple server deployments associated with TFS makes disaster-recovery planning and execution complex.

The Perforce Server architecture is much simpler. For core SCM functionality, the Perforce server software is self-contained, including an embedded database for tracking the metadata. Because Perforce uses standard file systems, network features, and a single server, the only overhead added is that associated with backup and regular user maintenance activities. Typically, one part-time Perforce administrator can support about a hundred users.

## Platform Support

Large enterprises usually consist of heterogeneous platforms, and the ability to manage projects using the same SCM tool is a requirement.

Team Foundation Server is a Windows-based product, with only a few third-party client interfaces available on non-Windows platforms. TFS does not fully support a 64-bit installation; while the data tier can run natively on 64-bit Windows, the application tier only runs on 32 bit Windows and does not even support WOW64 (for running 32-bit applications on 64-bit Windows).

The Perforce Server is available on a wide variety of platforms, including Windows, Linux, Mac OS X, Solaris, and UNIX. The Perforce server has native support for 64-bit operating systems, including Windows. P4V, the Perforce Visual Client, is a cross-platform client that is available on Windows, Linux, Solaris, Mac OS X, and FreeBSD.

## Third-Party Integration

Team Foundation Server is a core component of the Microsoft Visual Studio Team System (VSTS) product suite. While some third-party interfaces for TFS are available, it works best as part of the VSTS suite. Users of TFS are strongly encouraged to buy the entire VSTS suite, rather than using different products for different parts of the application development lifecycle.

Perforce is a comprehensive, best-of-breed SCM solution that integrates well with a large number of **third-party technologies**, including leading defect trackers, IDEs, and build tools.<sup>4</sup> Perforce's integration with Microsoft Office is better than Microsoft's; while TFS is limited to work-item integration with Microsoft Excel and Microsoft Project, Perforce provides a more robust integration—both for versioning and defect tracking—for not only Microsoft Excel and Microsoft Project, but also for Microsoft Word and Microsoft PowerPoint.<sup>5</sup>

Perforce's flexibility enables enterprises to preserve their investment in currently deployed tools and select from the large variety of products available in the marketplace that interoperate with Perforce.

## Support

Both Microsoft and Perforce Software provide global support. Because the quality of a technical support organization is better experienced than described, Perforce encourages its prospective customers to judge for themselves in a typical 45-day product evaluation.

Expert and responsive technical support is a hallmark of Perforce and full technical support is included during an evaluation.

## Additional Resources

Visit [www.perforce.com/trial](http://www.perforce.com/trial) for online SCM resources including:

- **Free Download:** Get a free 2-user, non-expiring copy of Perforce
- **Live Demo:** Register for a live weekly webinar or schedule a personalized demo
- **Product Overview Video:** Watch a video featuring the multi-platform PerforceVisual Client, P4V

To learn more about Perforce SCM, or to get SCM conversion resources, visit:

- **Agile & SCM:** To see how Perforce facilitates Agile development, go to: [www.perforce.com/perforce/agile](http://www.perforce.com/perforce/agile)
- **Conversion Resources:** For additional SCM comparisons, terminology, and conversion utilities, go to: [www.perforce.com/comparisons](http://www.perforce.com/comparisons)

- 
1. MSDN Library, "Team Foundation Version Control Scripts and Command Files," Microsoft.  
<http://msdn.microsoft.com/en-us/library/1az5ay5c.aspx> (accessed September, 2009).
  2. MSDN blog, "How to handle 'The path X is already mapped in workspace Y,'" Buck Hodges.  
[http://blogs.msdn.com/buckh/archive/2006/09/12/path\\_is\\_already\\_mapped\\_in\\_workspace.aspx](http://blogs.msdn.com/buckh/archive/2006/09/12/path_is_already_mapped_in_workspace.aspx) (accessed September, 2009).
  3. CodePlex, "Baseless Merges," Microsoft.  
<http://branchingguidance.codeplex.com/Wiki/View.aspx?title=Baseless%20Merges> (accessed September, 2009).
  4. Perforce Software, "Third-party Software Integrations," Perforce Software.  
<http://www.perforce.com/perforce/products/integrations.html> (accessed September, 2009).
  5. Wikipedia, "Microsoft Office Integration," Wikipedia.  
[http://en.wikipedia.org/wiki/Visual\\_Studio\\_Team\\_System#Microsoft\\_Office\\_Integration](http://en.wikipedia.org/wiki/Visual_Studio_Team_System#Microsoft_Office_Integration) (accessed September, 2009).



[www.perforce.com](http://www.perforce.com)

### North America

Perforce Software Inc.  
2320 Blanding Avenue  
Alameda, CA 94501  
Phone: +1 510.864.7400  
[info@perforce.com](mailto:info@perforce.com)

### Europe

Perforce Software UK Ltd.  
West Forest Gate  
Wellington Road  
Wokingham  
Berkshire RG40 2AQ  
UK  
Phone: +44 (0) 845 345 0116  
[uk@perforce.com](mailto:uk@perforce.com)

### Australia

Perforce Software Pty. Ltd.  
Suite 3, Level 10  
221 Miller Street  
North Sydney  
NSW 2060  
AUSTRALIA  
Phone: +61 (0)2 8912-4600  
[au@perforce.com](mailto:au@perforce.com)