|  |
| --- |
| 3M Corporate R&D |
| Enterprise Team Foundation Server Requirements |
| SEMS Lab |

|  |
| --- |
| Mike O'Brien  5-29-2014 |

Contents

[1 Document Description 2](#_Toc389211180)

[2 Overview of TFS and ETFS 2](#_Toc389211181)

[2.1 What is Team Foundation Server (TFS)? 2](#_Toc389211182)

[2.2 TFS Definitions/Terminology 2](#_Toc389211183)

[2.3 Why does 3M need an Enterprise Team Foundation Server? 2](#_Toc389211184)

[3 Roles 3](#_Toc389211185)

[4 System Interaction 4](#_Toc389211186)

[5 Functional Requirements 4](#_Toc389211187)

[5.1 Team Project Strategy 4](#_Toc389211188)

[5.2 Requirements Management within a Team Project 7](#_Toc389211189)

[5.3 Reporting 7](#_Toc389211190)

[5.4 Build Automation 8](#_Toc389211191)

[5.5 Version Control 9](#_Toc389211192)

[5.6 System Security 9](#_Toc389211193)

[5.7 Collaboration Portal (SharePoint) 12](#_Toc389211194)

[5.8 Support Portal 12](#_Toc389211195)

[6 Non-Functional Requirements 14](#_Toc389211196)

[6.1 Performance 14](#_Toc389211197)

[6.2 Monetization 14](#_Toc389211198)

[6.3 System Upgrades 14](#_Toc389211199)

[6.4 Backup and Data Retention 15](#_Toc389211200)

[6.5 Team On-boarding 15](#_Toc389211201)

[6.6 Disaster Recovery 17](#_Toc389211202)

[7 Governance 18](#_Toc389211203)

[7.1 Shared Service Security Planning 18](#_Toc389211204)

[7.2 Process Governance 18](#_Toc389211205)

[7.3 Build Service Governance 18](#_Toc389211206)

[7.4 Integration Governance 18](#_Toc389211207)

[7.5 Reporting Governance 18](#_Toc389211208)

[8 Abbreviations and Acronyms 19](#_Toc389211209)

# Document Description

Do we want to mention here the purpose of this doc (to give requirements) and the other docs that you’ve created (implantation guides, training, architecture docs, etc.)? That might clear up any questions right away.

Microsoft’s Team Foundation Server (TFS) is a world-class enterprise lifecycle management tool and can be leveraged by business groups to get projects to market quickly and effectively. This document outlines the requirements for the 3M enterprise-wide implementation of TFS thereafter known as Enterprise Team Foundation Server (ETFS) Service.

The requirements for ETFS presented in this document pertain to the core functionality of Team Foundation Server. Given that TFS is very flexible and customizable, these requirements define best practices and expected use of the system.

The requirements documented here are a result of feedback and interviews gathered from multiple teams and users of TFS and of the SEMS TFS server. Division participants include HIS, IPD, and current users of the SEMS system.

Abbreviations and acronyms are used extensively throughout the document and are defined in a list at the end of the document. Use cases to illustrate the concepts presented here are available in a separate document.

# Overview of TFS and ETFS

## What is Team Foundation Server (TFS)?

Microsoft’s Visual Studio Team Foundation Server is a collaboration platform that facilitates an application lifecycle management (ALM) solution. More information on the latest version can be obtained from the Microsoft product information website, currently at: <http://msdn.microsoft.com/en-us/library/ff637362.aspx>

## TFS Definitions/Terminology

* **Team Project:** Each individual project in TFS is known as a Team Project and has its own list of requirements, work items, source code, and team members.
* **Team Members:** Each member of the team will have one or more roles assigned which control the Team Project functions available him/her.
* **Team Project Collection:** A collection of Team Projects that groups Team Project together, and allows for sharing and querying of Team Project artifacts across Team Projects.
* **TFS Instance**: The server level – there is a single instance for each server deployed, with a plan for a single ETFS Instance.

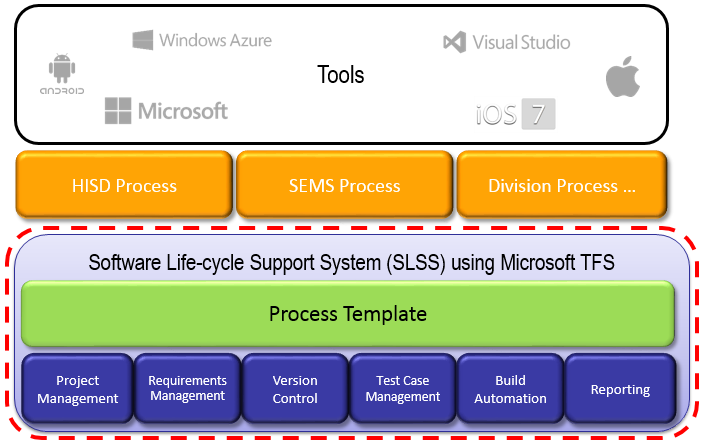
## Why does 3M need an Enterprise Team Foundation Server?

A recent assessment of multiple business groups within 3M concluded that:

* Independent work groups make use of a broad range of tools.
* The use of this broad range of tools results in technology silos.
* The tools are underutilized.
* Tools are not integrated, i.e., no Start-to-End integrated visibility.
* There is a lack of common infrastructure.
* Solution and projects are each built independently.
* There is little or no reuse of solutions across groups/divisions/company.
* There is a redundancy of personnel and roles across projects/groups/divisions/company.

Enterprise Team Foundation Server can help by:

* Providing a “One source of truth” repository for projects.
* Aiding in defining standard lifecycle and development processes.
* Reducing infrastructure support.
* Increasing the effective use of tools.
* Unifying the “Start-to-End” integration.



# Roles

There are many roles that make up a project team. The list below contains the primary roles on a team and the primary function for each, as they pertain to an ETFS project. As much as possible, these roles are abstracted from a specific software development methodology but, when necessary, Scrum terminology is used for clarity.

|  |  |
| --- | --- |
| Role | Primary Functions |
| Project Administrator (Project Manager/Scrum Master) | Create Team Project, manage Team Project permissions, enter Product Backlog Items (Work Items), organize Product Backlog Items with Iterations and Areas |
| Software Architect | Create team project architecture |
| Developer | Check in & check out source code, branch source code, |
| Business Analyst | Create application specification |
| Designer | Create/specify application interaction |
| Quality Assurance (Tester) | Create Test cases, create test plan, execute test cases, create bugs |
| Project Automation (DevOps) | Check in & check out source code, branch source code, create/edit build definitions, automate testing |
| Application Deployer (Operations) | Deploy application, create bugs |
| Project Sponsor (Stakeholder) | Provide Team Project guidance, prioritizes Product Backlog Items |

# System Interaction

The main system roles are represented in the system interaction diagram below. This is not all-inclusive, but illustrates the major interactions with ETFS.

Figure - System Interaction Diagram

# Functional Requirements

Functional requirements describe the installation, maintenance, customization and/or extension(s) of the ETFS service. The functional requirements are organized by the main aspects of TFS functionality, as shown below.

## Team Project Strategy

Each individual project in ETFS is known as a Team Project and has its own list of requirements, work items, source code, and team members. When a project begins to use ETFS either a new Team Project can be created or a team can join an existing Team Project. The team shall submit a request along with a list of required information about the project. The Team Project Strategy then dictates, based on this project information, when and where a Team Project is created, what type of process the Team Project supports, how security is structured, and the Team Project Collection direction and use.

### Project Administrator

The user of the system that initiates a request to create a new Team Project automatically becomes the Project Administrator. This is typically a Project Manager (or the Scrum Master) within a team, but any individual can request a new Team Project. The Project Administrator is charged with oversight of the Team Project, and is also the main contact for the Team Project should an issue arise.

### Team Project Creation

To create a new TP, information from the Project Administrator is required. A table below lists the information necessary with a description.

|  |  |
| --- | --- |
| Information Property Name | Description |
| Team Project Name | Name of your Team Project (must be unique) |
| Department Code | 3M Department Code of owner business group |
| Project Administrator | Team Project contact, responsible for Team Project oversight |
| Project Administrator Members | AD Group of Team Members with Project Administrator access (optional) |
| Contributor Team Members | AD Group of Team Members with Contributor access |
| Builder Team Members | AD Group of Team Members with Builder access |
| Build Services | Will a build machine be utilized? (Y/N) |
| Source Control Type | Which source control type, TFVC or Git |
| SharePoint Integration | Create SharePoint site for collaboration (Y/N) |
| Process Methodology | * Microsoft Visual Studio Scrum 2013 * Other |

The TP information gathered needs to be collected and stored with the TP. This information will be used to contact the Project Administrator when the need arises, e.g., system upgrades, system updates, maintenance windows, etc.

### Team Project Access Levels

ETFS requires the use of Active Directory (AD) groups for access control management. TP access levels are specified in the [Roles Section](#_Roles) above. The interaction diagram below shows how an AD group is leveraged across the primary subsystems within TFS.



Upon creation of a TP, the specified AD groups are assigned. The Project Administrator then will add team members to each of the access levels by leveraging AD. If the TP requires more ‘Team Groups’, the Project Administrator must create additional AD groups in provisioning the access levels.

### Team Project Collection Strategy

The Team Project Creation Strategy illustrates the decisions needed to determine if the new Team Project is created in an existing Team Project Collection, or if a new Team Project Collection should be considered.



Figure - Team Project Creation Strategy

The needs of each Team Project will be evaluated to determine if it can be created in an existing Team Project Collection or not as follows:

* If a TPC exists to suit the needs of the TP (i.e., a regulatory TPC already exists), it will be added to that existing TPC.
* If the new TP requires isolation from other TP’s due to a business need (e.g. regulatory and/or auditing purposes) and the TP security offered via the Areas is not enough isolation, a new TPC is recommended.
* If TP artifacts (source control, work item links) will be shared across multiple TP’s, or if the Process Template and security model are supported within the TPC, the current or same TPC will be used.
* Note: If the new TP requires very specific needs, i.e., isolation from other team projects and limited physical access to the hardware, a stand-alone TFS install that will be supported by the project team is recommended.

The ultimate goal is to keep the number of active TPs and TPCs to a minimum. The increased numbers of TPs and TPCs increases both administrative and maintenance costs.

## Requirements Management within a Team Project

### A Project Administrator can organize work items

A Project Administrator must be able to organize work items into segments of time (iterations) and into categories (areas) for organizational purposes.

### A User can assign a user to a work item

Any user with edit-project-level access can edit a work item and assign to an individual or group to perform the work. Provide usable solution to ‘Assign To’ list in work item assignment.

### A user can run Team Project level reporting to show project progress.

Any user with report-view-level access needs the ability to run reports against collection and tabulated information for a Team Project that shows project work completed and remaining work.

### A user can view supported Process Templates.

As part of communications with all users, both current and new, the supported process for team projects must be available for review.

### A user can create a new work item.

Any user with the appropriate permission shall have the ability to create a new work item.

### A user can update an existing work item.

Any user with the appropriate permission shall have the ability to update a work item.

## Reporting

The Reporting subsystem in TFS uses an aggregation method to build several layers of data for reporting.

Each TPC contains details of every transaction made by team members, such as source code Change sets, Work Item creation/modification, Team Builds, etc. The Tfs\_Warehouse aggregates this data from each TPC, and the data is further aggregated into Tfs\_Analysis.

### Initial Reports for Team Projects

Upon creation of the Team Project, an initial set of reports must be available to the project team. These must, at a minimum, include the following project status reports:

* Work Complete
* Work Remaining
* Testing Status
* Bugs/Issue Status

### Team Project Collection Reports

For a Team Project Collection, roll up reporting for Team Project across a Team Project Collection must be supported. This includes overview reports of key metrics for Team Projects as follows:

* Work Complete
* Work Remaining
* Testing Status
* Bugs/Issue Status

### TFS Instance Reports

Reports at the TFS Instance level are primarily focused on system performance, utilization, and oversight. Reports at this level must support:

* Tfs\_Analysis Cube Status
* Tfs\_Warehouse Job Status

As well as the following performance metrics:

* Execution Times
* Source Control Request Queue
* Users bypassing Proxies
* Historical Performance Trends across Work Items and Version Control
* Recent Performance Trends – This is granular data on recent performance usage across Work Items and Version Control.

## Build Automation

One useful component available with the ETFS Service is build automation. Build automation allows a user to make a request for an automated build of source code that resides within a Team Project.

### Build Machines - Enterprise

There must be a level of support for commonly used build types. The two configurations that are currently supported are:

* Latest version of OS - Windows Server 2012(R2) and version of .Net (4.5.1)
* Previous version of OS - Windows Server 2008 R2 and version of .Net (4.x)

### Build Machine – Dedicated

ETFS must support build machines that may include customized dependency installs of software both on premise (data center) and on a remote site.

As software complexity increases, the installation of dependencies on the build machine is becoming more common. A method of handling these workflow issues is critical for adoption.

### Support for Non-Windows builds (e.g., Xamarin, iOS, Java).

Divisions are currently working on products across multiple platforms. EFTS must provide an automated build system capable of handling:

1. Windows/.Net
2. iOS
3. Java
4. Xamarin (<http://www.xamarin.com> – a cross platform development environment)

## Version Control

Revision control of source code and project assets are a critical function of any ALM solution. The following requirements outline basic version control abilities that must be implemented in ETFS.

### Functionality

Basic tenants of source control need to be supported as part of the ETFS system including the check-out, check-in, locking of project assets, branching, and merging.

### Client Support

The source control system will work with current development environments, namely:

* Visual Studio 2010 / 2012 / 2013
* XCode
* Eclipse

### Retention

When a software team performs a release, the source code can be archived and stored in the source control system. The team must have the confidence to know the source is secure and stored for future examination.

A team can perform maintenance on the released version of project~~s~~ source code after a major release. The functionality to perform maintenance on a release months or even up to 3 years after release must be provided.

### Source Control Support

A user can use any form of Source Control supported by TFS within ETFS. The current forms of source control supported in the 3M ETFS implementation include:

* TFVC (Team Foundation Version Control)
* Git

### External / Outsourced Development

Division customers would like the ability to provide an outsourced company with a software development tool to view project progress and review source code. *These requirements need to be explored further with regard to the Master Service Agreement and security requirements from IT Security.*

#### External / Outsourced software development needs accessibility to a 3M Software Development System.

Ideally, an external agency would be provided access to an ALM solution for project development.

#### The externally available system can handle outsourced companies without Active Directory domain accounts.

In addition to the previous requirement, the externally available system would be independent of Active Directory authentication.

## System Security

Defining Team Roles and providing role-based security is one key to a well-run project. The definition of role-based authorization within ETFS is described for each of the three tiers of operation – TFS Instance, Team Project Collection, and Team Project. A hierarchy of permissions exists within every Team Foundation Server, as shown below.

Figure - ETFS Tiers

Permissions are detailed in the sections below including access levels at the differing ETFS Tiers and the authorization levels.

### Team Project Security

A Team Project has several levels of necessary Team Project Roles, or access levels.

|  |  |  |
| --- | --- | --- |
| Team Project Role | | Description |
| Project Administrators | Project Level Administrators with full access to the Team Project. | |
| Builders | Can create, edit, and delete build definitions | |
| Contributors | Can create & edit work items, read & write source files | |
| Observers | Read only access to Team Project work items and source control | |

Each of these roles also correlates to the access levels in SharePoint and Reporting Services.

|  |  |  |
| --- | --- | --- |
| Team Project Role | SharePoint Access Level | Reporting Services Access Level |
| Project Administrators | Full Control | Team Foundation Content Manager |
| Builders | Design | Content Manager |
| Contributors | Contribute | Content Manager |
| Observers | View Only | Browser |

The access levels for SharePoint and Reporting Services is defined in following sections.

### Team Project Collection Security

A Project Collection Administrator group is needed at the Team Project Collection level. Members of this group have full access to all aspects of the Project Collection only.

|  |  |
| --- | --- |
| Project Collection Role | Description |
| Project Collection Administrators | Project Collection Level Administrators with full access to the Team Project Collection. |
| Project Collection Build Administrators | Can create, edit, and delete build definitions |

A Project Collection Build Administrator is needed to manage build services across a Project Collection.

These roles also correspond to the access levels in SharePoint and Reporting Services but, to a large effect, administrators in the Team Project Collection are Administrators in SharePoint and Reporting Services. Unique circumstances may allow for a divergence from this on a case-by-case basis.

### ETFS Instance Security

At the ETFS Instance level (or server level), the Team Foundation Administrators group is necessary to maintain an administrative level of the system. Members of this group have full access to all TFS services on the TFS servers.

A group to manage TFS instance service accounts is also necessary. Members of this group are a NPSA (Non-Person Specific Account), and used in operation of the system. This group is also a member of the TFS administrators group.

|  |  |  |
| --- | --- | --- |
| ETFS Instance Role | | Description |
| Team Foundation Administrators | Administrators with full access to the TFS Instance | |
| Team Foundation Service Accounts | Can create, edit, and delete build definitions | |
| SharePoint Web Application Services | Service accounts used by SharePoint to integrate with the TFS Instance | |

These roles correspond to the access levels in SharePoint and Reporting Services, but to large effect, administrators in the ETFS Instance are Administrators in SharePoint and Reporting Services.

### Reporting Access Levels

Three levels of access control are necessary to support the different levels of running reports as follows:

|  |  |
| --- | --- |
| Reporting Access Level – Role Name | Permission to |
| Team Project Reporting | Run reports across the Team Project |
| Team Project Collection Reporting | Run reports across a Team Project Collection – view data from multiple Team Projects |
| TFS Instance Reporting | Run reports across the TFS Instance – server level access to view all Team Project data |

A user with Team Project Reporting access has the ability to run reports against a Team Project.

A user with Team Project Collection Reporting access has the ability to run reports against a specified Team Project Collection.

A user with TFS Instance Reporting access has the ability to run reports against collection and tabulated information for all Team Project Collections.

The [System Security](#_System_Security) section below defines roles necessary for reporting in the form of AD groups for a team project. Team Project Collection Reporting and TFS Instance Reporting level Active Directory groups must be maintained independent of the Team Project Active Directory groups by the TFS Administrators team.

### Collaboration Portal Access Levels

A Collaboration Portal (SharePoint) needs the following access levels to accommodate the varying levels of functionality and control of project artifacts.

|  |  |  |
| --- | --- | --- |
| SharePoint Access Level Role Name | Permission | Default SharePoint Access Level |
| SharePoint Administrator | Full Control of the Team Project SharePoint Portal. | Full Control |
| SharePoint Builders | Read and Write permissions with limited Administration abilities | Designer |
| SharePoint Contributors | Read and Write permissions to the Team Project SharePoint Portal | Contribute |
| SharePoint Readers | Read-only access to the Team Project SharePoint Portal and contained project artifacts. | View Only |

### The security model for the system must be documented and shown to users.

The security model and overall structure of the system security should be available to all users of the service. Administration, security groups, and access levels must be communicated to users.

### Active Directory must be used as the source of user authentication.

Active Directory is the current 3M standard for desktop and Windows Server authentication, and must be leveraged for ease of maintenance.

The system should be implemented in accordance with the best practices for TFS in regard to security and account structure.

Industry best practices should be followed when valid. Service accounts must be created and used in accordance with the best practices of TFS. These best practices are analyzed by use of the ‘Microsoft Team Foundation Server Best Practices Analyzer”.

## Collaboration Portal (SharePoint)

### A SharePoint collaboration portal must be available to teams.

### A master page that provides functional information and current status of team progress must be available to project teams.

## Support Portal

ETFS will support two methods of reporting system issues and getting support

* A user can call 3-1000 or go to ithelp.mmm.com and report the issue. A support ticket will be assigned to the **US\_TFS-App-Support** ITSM support group.
* A user can go to the ETFS Support Portal and report the issue.

The Support Portal offers the functionality described in the next several sections:

### A user can schedule training with the support team.

Any user can schedule the training, but the preference is a Project Administrator for a Team Project to request team training.

### A user can view the recent service changes and modifications.

The list of service changes and modifications is open to any user.

### A user can view the current status of all running Services (dashboard).

The list of Services and current status is open to any user.

### A user can view service policies.

Any user can view service policies for the system:

* Description of ETFS Service and available services.
* Build Services Available – including team build retention
* All Governance policies listed below.

### A Project Administrator can request a custom report to be created and/or modified.

Customizations to a Team Project report can be accommodated if they do not conflict with the Reporting Governance.

### A Project Administrator can request customizations to their TP process template.

Customizations to a Team Project process can be accommodated if they do not conflict with the Process Governance.

### A Project Administrator can view current monetization for their Team Project.

The Project Administrator needs to view current monetization owed for appropriate team projects they have ownership for. See the [Monetization](#_Monetization) Section below for more details.

### Received support requests must be started within 24 hours of receiving.

Critical issues must be addressed quickly by the support team, with a goal of 24 hours to close.

### Received support requests must be addressed in a timely manner.

Support requests should be addressed as quickly as possible, with a goal of addressing the request within 3 days of receiving.

# Non-Functional Requirements

The non-functional requirements are based on customer feedback and current best practices for implementation of an Enterprise level service within 3M.

## Performance

### The system needs to handle user load with redundancy.

System performance needs to be monitored, and kept at an acceptable level of service, i.e., connections to the service must not time out or return errors.

### The system will be highly available with an expected uptime of 99.9%.

The system must be highly available, and provide an expected update of 99.9%. This is not inclusive of scheduled maintenance windows and scheduled downtime.

### Remote Sites

Division customers currently have remote offices with slow network connections to the St. Paul data centers. As much as possible, the slow connections need to be mitigated with TFS Proxy machines. Each remote site will be onboard-ed on a case by case basis, and have the remote connection speed assessed when an ETFS Team Project is requested.

## Monetization

Monetization, or chargeback to customers, may be necessary to support the ETFS Service. A decision on Monetization has not been made, and will be investigated in a Pilot of the ETFS Service with a division customer. The pilot will gather data to detail the level of support necessary to keep a project team successful, which in turn gives a level of effort. Monitoring usage and/or number of active users are 2 potential methods of monetization, in addition to the support hours.

### Usage Monitoring

To determine a level of chargeback, these characteristics will be monitored for usage:

* Web service calls or system utilization
* Storage usage
* Build machine usage - # of minutes performing builds

### Active Users

To determine a level of chargeback, the number of active users assigned to Team Projects and Team Project Collections will be tabulated on a monthly basis.

## System Upgrades

The diagram below shows a high level process for the upgrade to a new version of TFS.

This process will be performed for the TFS Instance, with each TPC being upgraded.

This process is a recommendation, and is open to modifications based on updates, new functionality, and features added to the new release of TFS.

The timing of when an upgrade is performed is under the governance of the ETFS Steering Committee.

### A TFS version upgrade can be performed within the defined service window.

As suggested by the TFS Planning Guide, schedule resource increases for upgrades (CPU, Memory). Upgrades to the ETFS system will be posted on the main ETFS website. Team Project administrators will be contacted 30 prior to system upgrades.

### A TFS release update can be installed within the defined service windows.

Updates to the ETFS system will be posted on the main ETFS website. Team Project administrators will be contacted 30 prior to system updates.

## Backup and Data Retention

Backups of data must be performed in accordance with the Global Standard Operating Procedure – found at <http://3msource.mmm.com/wps/myportal/3M/en_US/About-3M-Policy-Center/WW-corporate/infotech/?univid=1319221135500>

## Licensing

The ETFS system must be licensed properly and in accordance licensing standards for products used.

## Team On-boarding

To achieve a high level of adoption and acceptance of the ETFS Service, a defined process must be created to onboard new teams onto ETFS. This process also needs to be reevaluated monthly and adjusted accordingly.

### On-boarding Process

The Team on-boarding process follows the stages in the diagram below.

#### Initial Meeting

Part of the initial meeting will include a presentation of the capabilities of Team Foundation Server / Visual Studio. There is the potential to have Microsoft co-present.

#### Team Interview

* Collect information on team(s)
* TPC / TP – follow Team Project Creation workflow to determine recommendation on Team Project location
* Follow On-boarding WF for Work Items and Source Control
* Migrate artifacts vs. archive
* Team Size
* Methodology
* Connection speed (to mitigate “slow” remote connections)
* Large Project?
* 100+ users
* Complex needs



Figure - Source Control and Work Items Onboarding Workflow

#### Training

Training must be available to teams as part of the on-boarding process.

Multiple levels of training should service the differing levels of functionality of teams being on-boarded including the following:

* Introduce Team Foundation Server & ETFS
* Identify Team Roles
* Create training curriculum based on desired Usage Scenario
* Train in Context by demonstrating key tasks
* Provide access to Hands-on-Labs
* Provide access to Trial instance of ETFS

#### Migration

* Trial migration performed by ETFS team, if necessary, develop solutions
* Build and Proxy machine provisioned and connected to ETFS

#### Go Live

* Team(s) switches to using ETFS on scheduled date/time.
* Archive system put into read-only mode

#### Maintain relationship throughout project

* Are needs being met? Does ETFS provide the necessary functionality?
* Projects running smoothly?
* Training helpful? Anything missing?
* Methodology working?

### If required by the business, migrate source control into TFS with as high of fidelity as possible (full history)

### If required by the business, migrate Work Items into TFS with as high of fidelity as possible (full history).

## Disaster Recovery

Disaster Recovery plans must be created to cover recovery from known issues that may arise on the ETFS Service. Outlined below are Disaster Recovery plans that must be created and tested.

### Avoidance

“See the smoke before the fire.” Monitoring of the ETFS Service is the primary means of avoidance of Disaster Recovery. See the section describing the [Non-functional Requirements](#_Non-Functional_Requirements) for more details. However, even the most pro-active and best detective measures cannot always guard against man-made or natural disaster.

### Planning

Information about preparing ETFS for disasters, which is essential to recovering from a disaster, must be shared in an effective and timely manner.

### Complete Failure (fire, natural disaster)

The system must provide the means to recover from a complete failure of the ETFS Service.

### Data Tier (DT) failure

If the Data Tier fails, outline the plan to recover.

### Application Tier (AT) failure

Recovery plan if an Application Tier machine fails.

### Proxy failure

Recovery from a failing Proxy machine.

### Build failure

Recovery steps when a build or build machine fails.

### SharePoint failure

Mitigation of a SharePoint server not functioning.

### Switch-over to secondary site

Direct users to a new secondary site until the primary site is functional.

# Governance

Governance of the system will be maintained by an ETFS Process and Governance Steering Committee. This group will meet regularly, and include representation from each Team Project.

## Shared Service Security Planning

A Shared Service Security Planning Governance policy must include a set of standards used to evaluate teams and provisioning the teams by assigning a container and security model based on various features to separate the teams such as TFS Instance, Team Project Collection, Team Project, or Area.

## Process Governance

A Process Governance policy must include a standard for managing Process Templates and customizations to both the Supported Process Templates and an individual Team Project, such as, adding a custom field.

## Build Service Governance

The Build Service Governance must include:

* An approach that can scale to the Enterprise
* Inclusion of a shared build environment that can be used by teams that have generic needs as well as a governance model that allows teams to that have unique needs to manage their own build server farms while still using the enterprise TFS Service in a standard way

## Integration Governance

The Integration Governance must include guidelines and standards to facilitate various integration needs that involve TFS, e.g., HPQC, JIRA, MS Project Server, etc.

## Reporting Governance

The Reporting Governance must include guidelines and standards for managing TFS Reporting and customizations.

# Abbreviations and Acronyms

|  |  |
| --- | --- |
| Abbreviation | Full Name/Description |
| AD | Active Directory |
| ALM | Application Lifecycle Management |
| Eclipse | Development environment (<http://www.eclipse.org>) |
| ETFS | Enterprise Team Foundation Server |
| Git | Distributed peer-to-peer source control system - |
| HIS | Health Information System (Division) |
| HPQC | HP Quality Center |
| IPD | Infection Prevention Division |
| JIRA | A [proprietary](http://en.wikipedia.org/wiki/Proprietary_software) [issue tracking](http://en.wikipedia.org/wiki/Issue_tracking_system) product developed by [Atlassian](http://en.wikipedia.org/wiki/Atlassian). |
| SEMS | Software Electronic Mechanical Systems (Division) |
| SSRS | SQL Server Reporting Services |
| TFS | Team Foundation Server |
| TFVC | Team Foundation Version Control |
| TP | Team Project |
| TPC | Team Project Collection |
| TWA | Team Web Access |
| Xamarin | http://www.xamarin.com |
| XCode | iOS development environment |