From	Our Reference	Tel	Coimbatore
RBEI/BSC	Sunil Kumar Rana	+91(422)673-2067	29 July 2014

ETAS TFS Transition Document

RBEI/BSC

Index

1 Objective

The objective of this document is to describe record and put down all the knowledge gained during the transition of ETAS TFS Topic from ETAS to RBEI.

2 Header Information

Name of Document	ETAS_TFS_Transition_V1.0.docx
Prepared by	Sunil Kumar Rana(RBEI\BSC2-TDP)
Reviewed by	Frank Beckmann (ETAS/ICO)
Duration of Transition	2 weeks i.e. 28-07-2014 to 09-08-2014
End User	All users working with ETAS TFS Tool for carrying out their project related activities

3 Revision History

Issue	Version	Date	Editor	Description of amendment
1	1.0	29-07-2014	Sunil Kumar Rana	Initial
2	1.1	30-07-2014	Sunil Kumar Rana	Added review points
3	1.2	08-07-2014	Sunil Kumar Rana	Added Build Server, Build Configuration at client side, and other build definitions.

4 Servers at ETAF TFS

Recently, all the servers from ETAS has migrated to Bosch as part of OneIT2015 Migration

Server Name	Category	Username
FE0VM736.de.bosch.com	Production	DE\ETR6FE
FE0VM661.de.bosch.com	Development	DE\ETF5FE
SI0VMC0201.de.bosch.com	Build Production	Own Admin Account
SI0VM1024.de.bosch.com	Quality/Test	DE\SVT5FE
SI0VMC0437.de.bosch.com	PreMigration2013	Own Admin Account
FE-Z0PTL.si.de.bosch.com (VDI)	Development for Visual Studio 2012	Own office account

Complete details of server side components are described in Section 6.

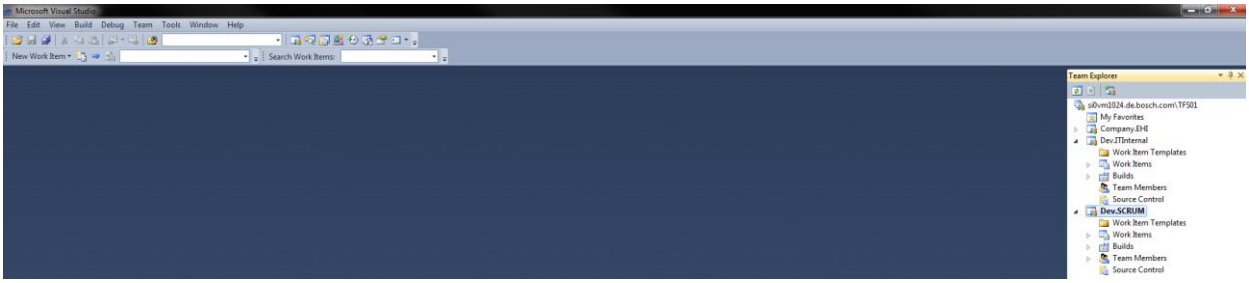
5 ETAS TFS Client Side Components

5.1 Introduction

Microsoft has provided two tools to connect to ETAS TFS Server i.e. via different Visual Studio Editions and Web Access.

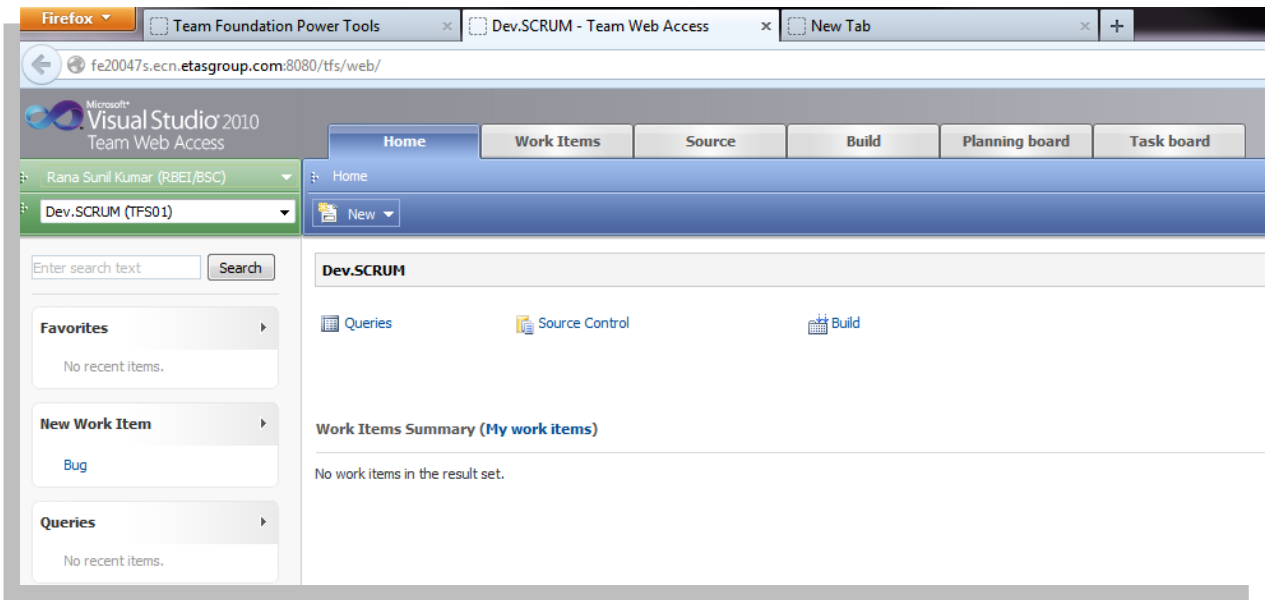
Microsoft Visual Studio is used as standalone client side component in connecting to ETAS TFS Server. Under Visual Studio, we have Team Explorer which acts as an interface in connecting to different Team Foundation Servers.

Below is a screenshot of Visual Studio Shell with Team Explorer as a client side component



The other way to connect to ETAS TFS Server is Web Access. As the word suggests, it is accessible only through web browser by providing URL as an input.

Below is a screenshot of ETAS TFS Server from TEST Environment



As you can see in the browser, the URL belongs to a test environment

All the TFS Client side cache would be available in the local cache directory of the user's machine.
Local Directory would always be: %APPDATA%\Local\Microsoft\Team Foundation

In the above folder, Caching is available with Host ID as a folder in GUID Format. It is very important to the category of environment that the cache belong to.

- For this, Firstly login into database either Prod or Test.
- Then, Select the first 1000 rows from the table as shown in the box

```
/****** Script for SelectTopNRows command from SSMS  ******/
SELECT TOP 1000 [ProcessId]
      ,[HostId]
      ,[StartTime]
      ,[Status]
  FROM [Tfs_Configuration].[dbo].[tbl_ServiceHostInstance]
```

- The Column ***HostID*** will display all the Hosts connected to the Server either prod or test.
- In this way, we can identify from the cache, which server the user is connecting to.

In case of any local client side issues in Visual Studio Tool, It would be advisable to clear off or delete only the contents of the Cache directories under 2.0 and 3.0 of local cache folder.

Along with ETAS TFS Client side components, ETAS has customized and added other features in the TFS Client side components to allow their users to be able to run their project related activities without any hassle.
ETAS has introduced and developed 4 major client side components which can be installed along with Visual Studio for users to work more effectively and productively.

They are as follows

- ETAS TFS Custom Controls
- ETAS Attachment Management Control
- ETAS OWI
- ETAS Search Work Items

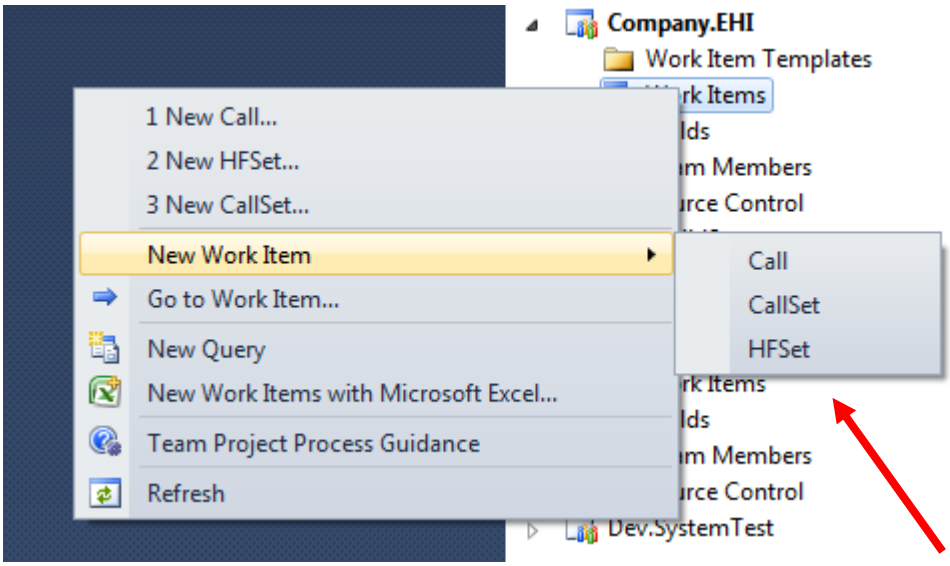
5.1.1 ETAS TFS Custom Control

ETAS TFS Custom Control is specifically designed for users who are working with *Company.EHI* Project under Project Collection *TFS01*

EHI stands for ETAS Helpdesk International.
Company.EHI is a ticketing system available under Visual Studio to create the tickets. It is raised by Support team/SAS Team on behalf of customer which will be then handled by appropriate team.
The available work item type under Company.EHI is Call, CallSet, HFSet.

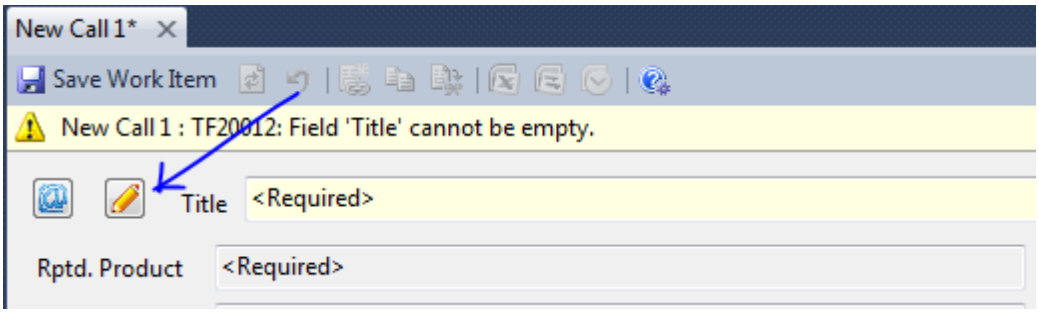
For know-how on how to connect to Company.EHI, You can refer to Section 5.2.1.

A call is a ticket raised to record, analyse the complaint/issues of customer, testers, developers, product managers related to a specific product
A Callset is a list of calls that can be labelled under a release by Project manager. All these calls will be available as part of this release in form of a Callset.
An HFSet is a hot fix call raised for emergency patch fix for customer.
Below is a screenshot of work item types under Company.EHI



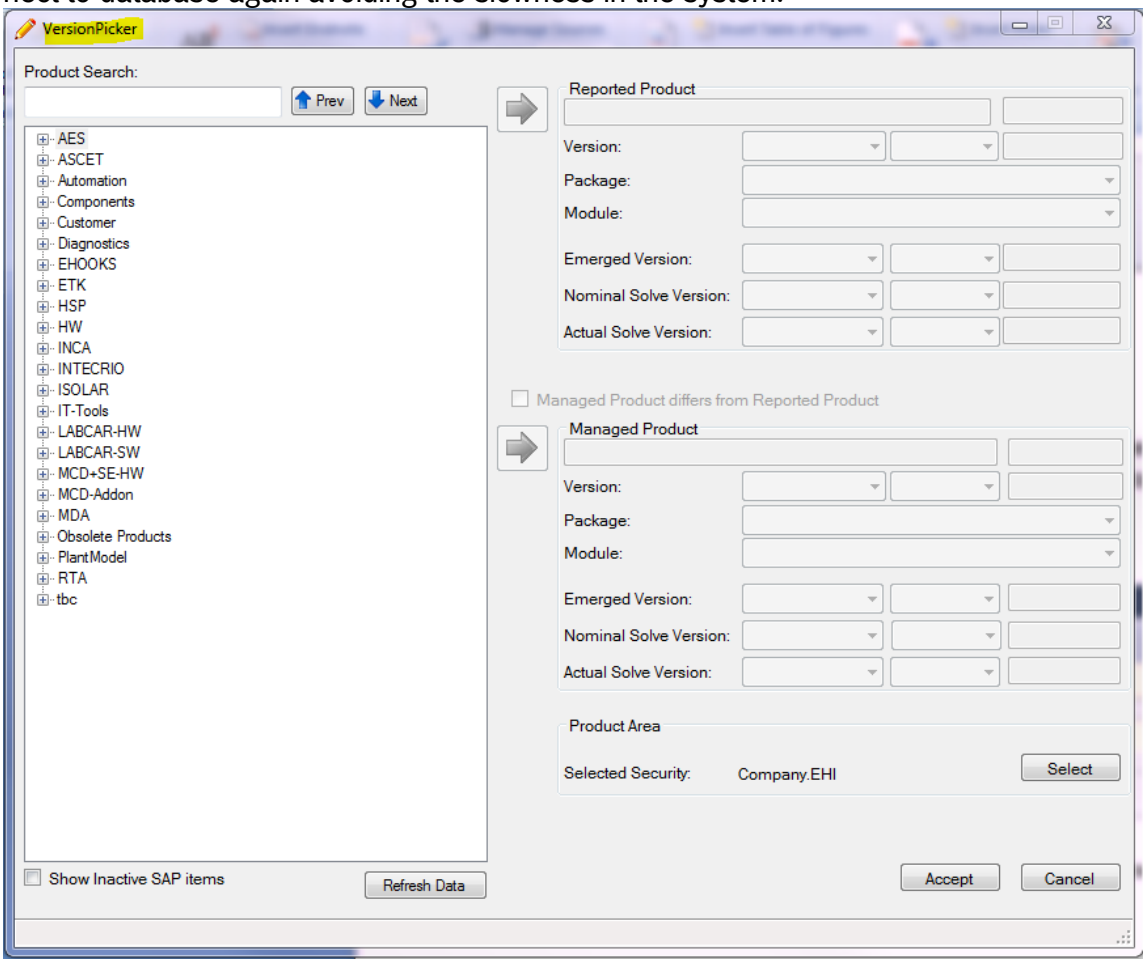
Now ETAS TFS Custom Control as a feature is available as a version picker under any work item type raised in Company.EHI. It is basically an icon on the top-left corner of the ticket.

Below is a screenshot of the ETAS TFS Custom Control Icon. Installables are available in the shared folder \\fe13606.ecn.etasgroup.com\csw_Standard_Software\Development\EHI-TFS2010Client#FREE_LIC\iCat_Version_TFS_2010Client\source\ETASTFSCustomControls\10.0



Upon clicking the version picker, we can see the another window popping something like below
All the data in the version picker would be collected from the %APPDATA%/Roaming/EHI-Cache

There will be 2 XML files which would act as data in the cache without the Visual Studio to connect to database again avoiding the slowness in the system.

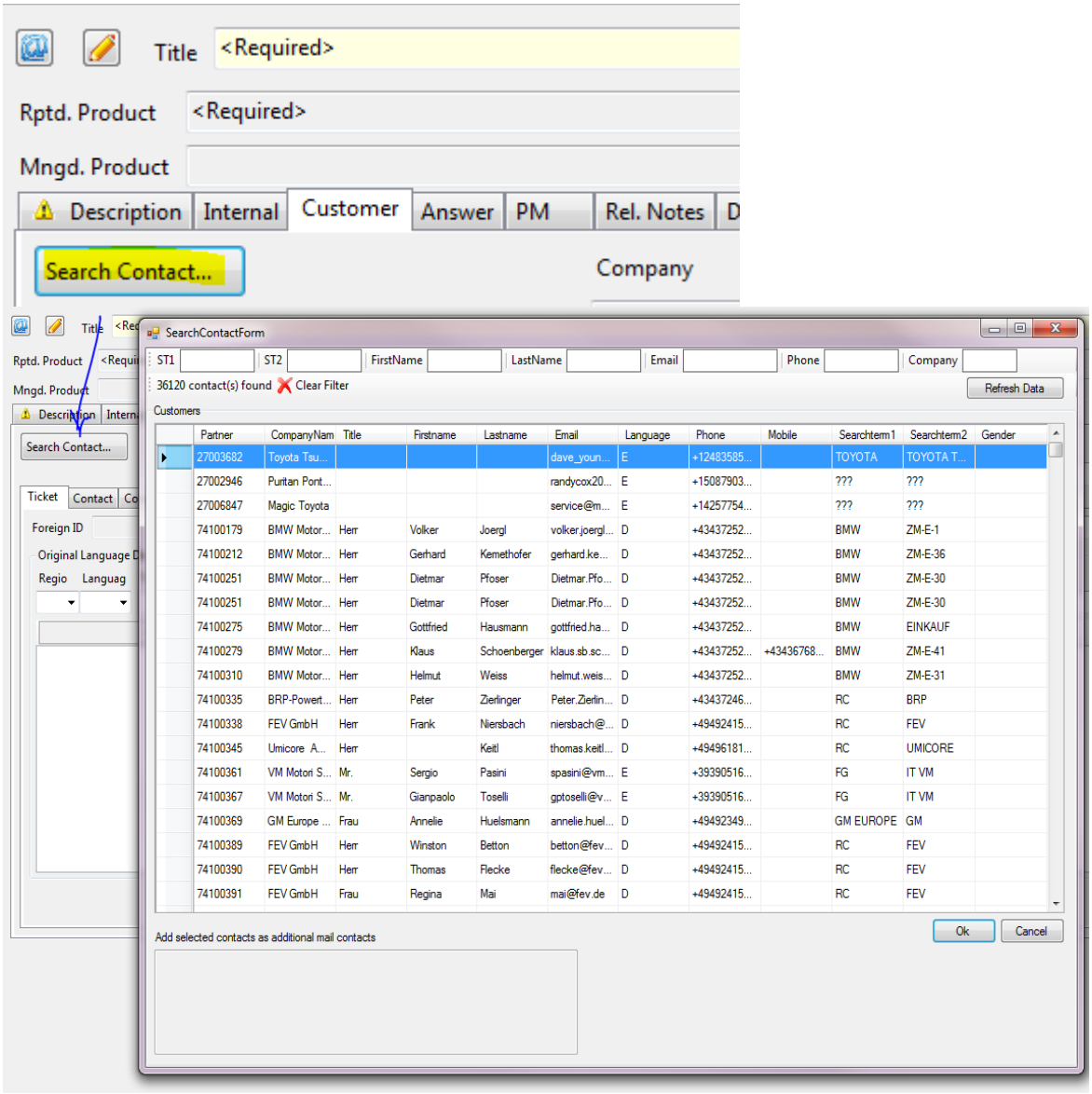


Now, The Requestor can fill the required fields and make it easier to put all the relevant information in the ticket.

Also, User can click on Refresh Data on the bottom of the screen to allow the Visual Studio to collect the data from the database instead of the cache. There would be slowness in the system.
A point to remember would be that Opening and Closing the application would collect the data from the database rather from the cache.

Other features also includes

- ***Search Contacts***
  Search Contacts will always display the latest list of the entire customer from the TFS database (TFS_EHI_ALL) which actually pulls the data from SAP System having the recent data through a web service running on a TFS Server.
  Below is a screenshot of the Search Contacts from Company.EHI under Call Ticket.



5.1.2 ETAS Attachment Management Control

ETAS Attachment Management Control is a feature developed by ETAS to enable the attachments to be stored on a file server rather than database. The aim is to avoid database as a storage device to help database growth minimal.

Installables are available in the folder
\\fe13606.ecn.etasgroup.com\csw_Standard_Software\Development\EHI-TFS2010Client#FREE_LIC\iCat_Version_TFS_2010Client\source\ETASTFSCustomControls

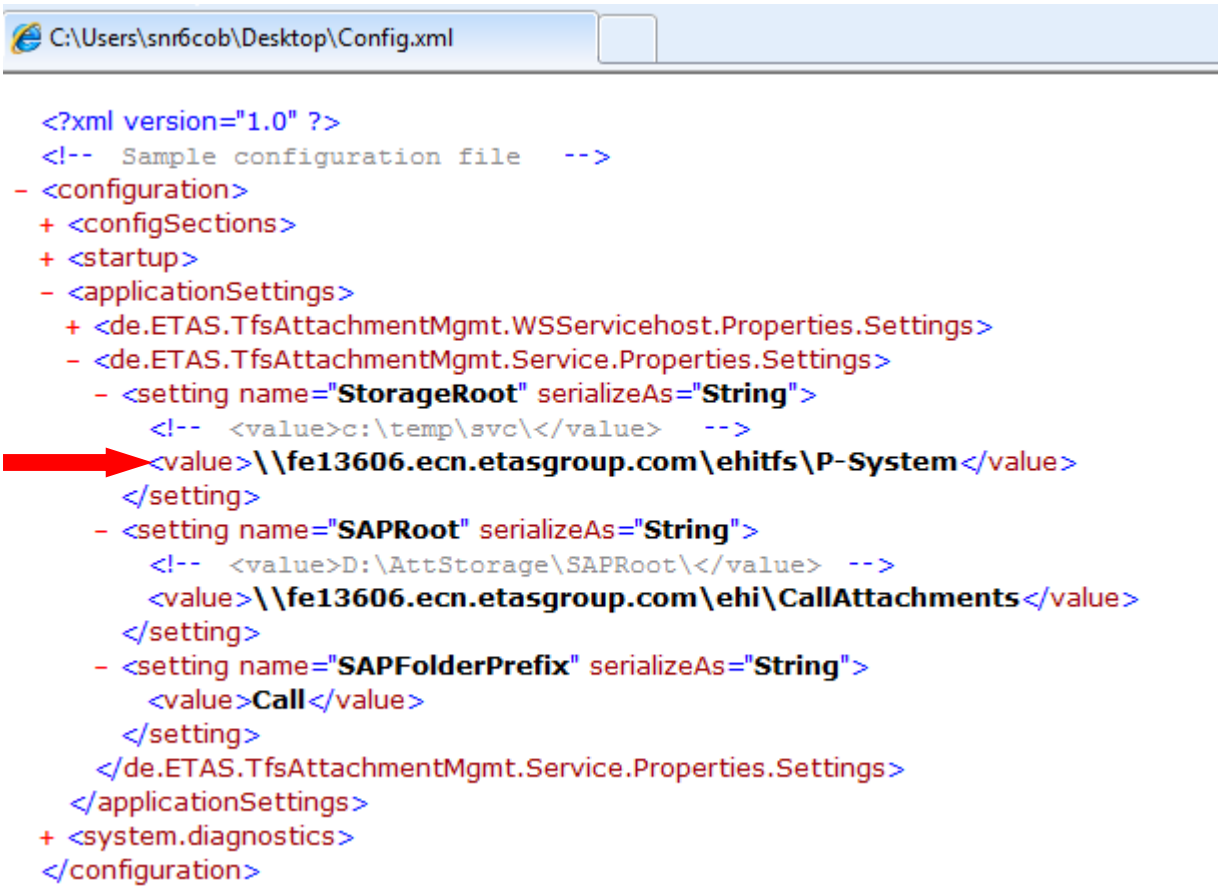
For both the ETAS Prod and Quality, the file server is different and is stored in two different locations.

To find out the location of the file server of both the ETAS TFS Prod and Quality Servers, You can refer to the config file being used by the ETAS Attachment Management Control on the servers respectively.

The location of the config file "de.ETAS.TfsAttachmentMgmt.WSServicehost.exe.config" is available in

C:\Program Files (x86)\artiso Solutions GmbH\AttachmentMgmt Server

If you open the file as an xml file, then you would see the tree structure as it is available in the XML File.



The value that has been provided to StorageRoot wil be the database of all the attachments being uploaded in Visual Studio Tool. This is basically used for Company.EHI Project.

5.1.3 ETAS OWI Setup

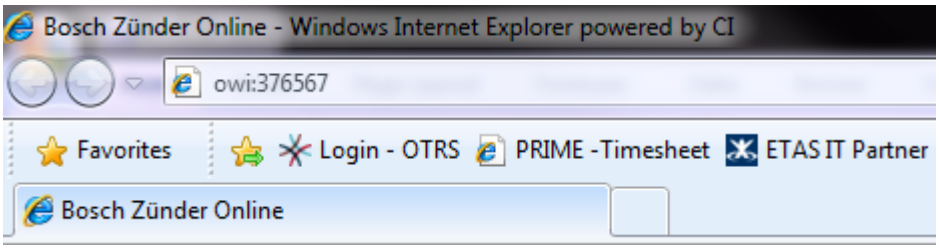
ETAS OWI is another feature available as part of ETAS Customized Tools for ETAS Users to work more effectively and productively with their project activities.

Installables are available in the folder
\\fe13606.ecn.etasgroup.com\csw_Standard_Software\Development\EHI-TFS2010Client#FREE_LIC\iCat_Version_TFS_2010Client\source\ETASTFSCustomControls

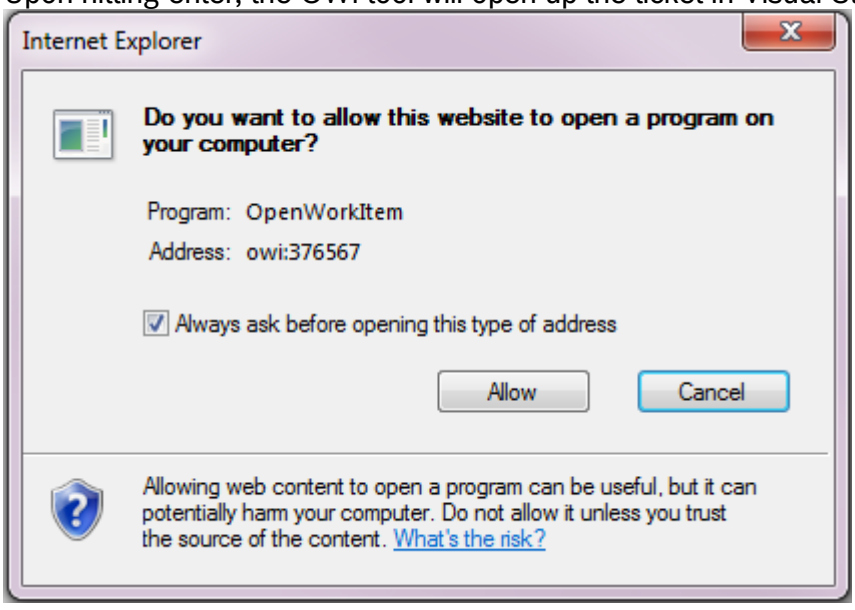
OWI Stands for Open Work Item and this tool can be used to open the work item directly from your web browser or windows explorer in a Visual Studio Tool.

All you need is to know the ID or the Open Work Item number and you can open up the ticket using OWI in Visual Studio Tool either from the web browser or the windows explorer.

```
owi:<Ticket Number>
```



Upon hitting enter, the OWI tool will open up the ticket in Visual Studio Tool



Click on Allow.

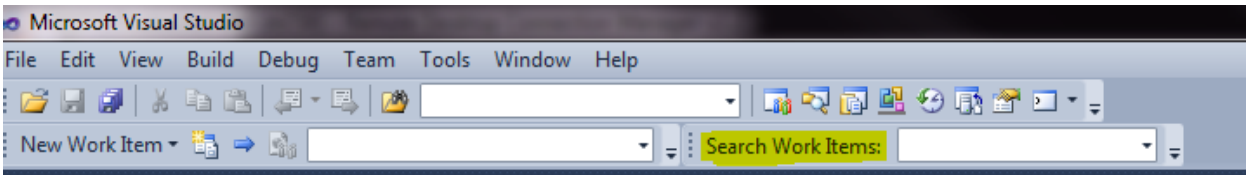
It will open up the ticket in a Visual Studio platform.

5.1.4 ETAS Search Work Items Setup

ETAS Search Work Items Setup is another feature provided by ETAS to search the work items directly from the Visual Studio Client side component.

Installables are available in
\\fe13606.ecn.etasgroup.com\csw_Standard_Software\Development\EHI-TFS2010Client#FREE_LIC\iCat_Version_TFS_2010Client\source\ETASTFSCustomControls

Below is a screenshot showing the ETAS TFS Search Work Item tool available on top of the Visual Studio Tool.



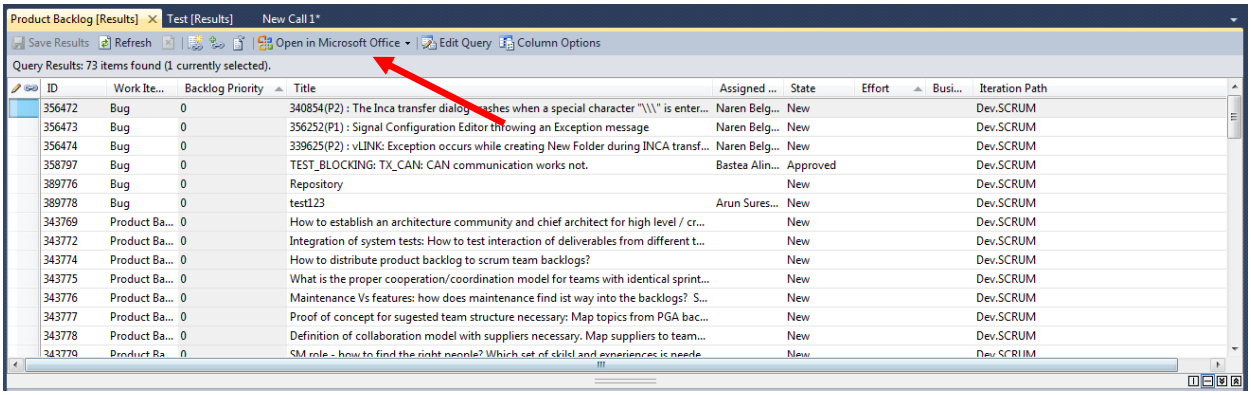
You can search for work items from your Visual Studio Tool.

5.1.5 TFS Publishing using MS Office Applications

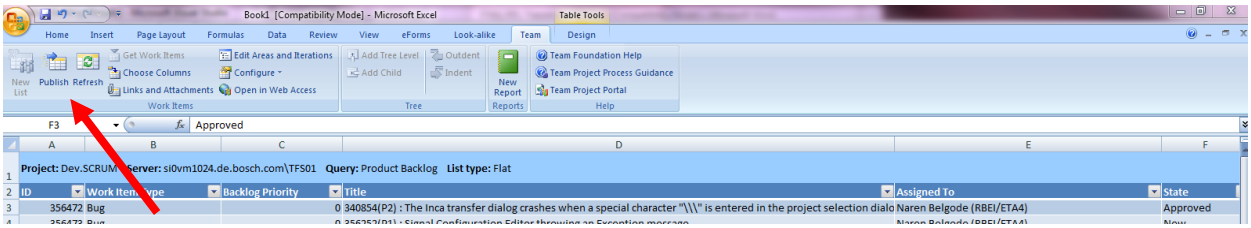
Team Foundation Server also gives an additional facility in Visual Studio Tool to make the changes to a list of call items in a single shot.

For example:

- We have a list of calls as shown in the below screenshot.



- Find an option called “Open in Microsoft Office”. You can select the options available as per your request. In our case, it would be the Microsoft Excel.
- After Excel shows up, you would find the list of tickets exported to excel document with publish button on top left of the document.



- Now click on “Publish” after you are done with changes. This would reflect in the list of calls in the Team Queries under Team Explorer.
  Note: Sometimes you need to refresh the work items list to see the updated changes.

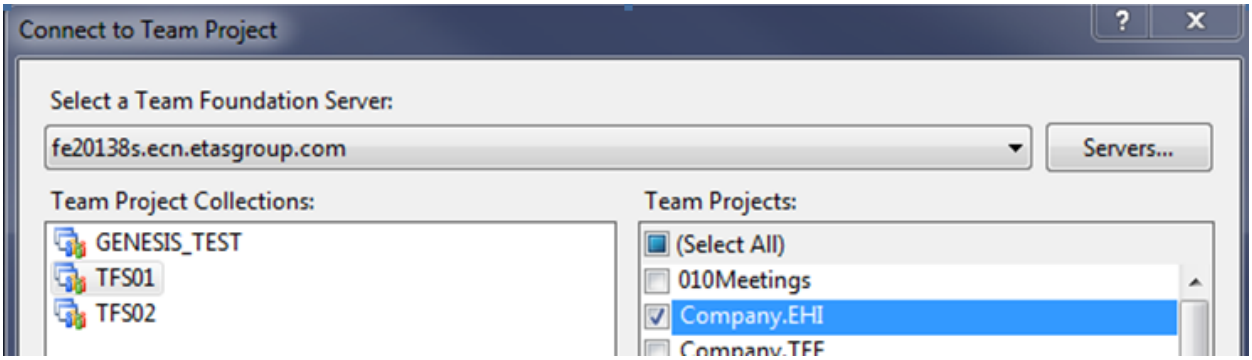
5.2 ETAS Client Side Applications

ETAS have several client side projects/applications under a Project Collection. Here are couple of one which are really important for ETAS from business perspective.

- Company.EHI
- Urban Turtle

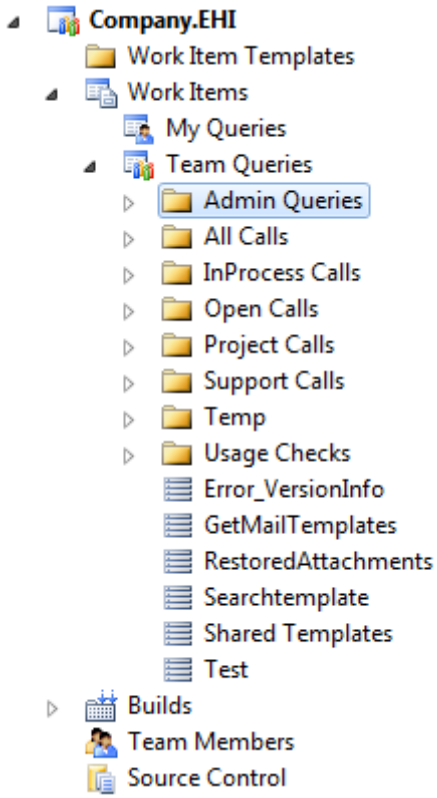
5.2.1 Company.EHI

Company.EHI is already discussed in Section 4.1.1. Please refer to the section to find more details.
Company.EHI is a ticketing system that enables the support team to raise tickets on behalf of customer.
Company.EHI is available under Project Collection TFS01 in the Prod Serve Fe0vm736.



After seeing the above window, You will see the connect button to connect to Company.EHI in Visual Studio.

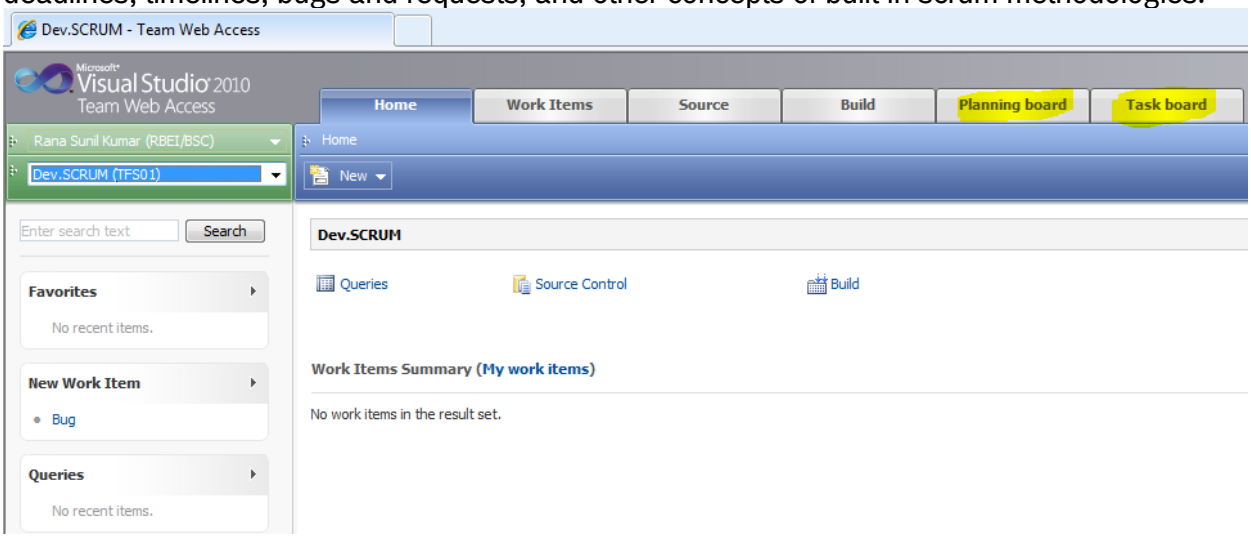
Below is a screenshot showing the contents available under Company.EHI. We can see Work Items, Team Queries, Builds, Team Members, and Source Control.



5.2.2 Urban Turtle

Well, Urban Turtle is an external application developed by PyXis. This is an ADD-ON to already existing Web Access of TFS Client Component. This gives the addition feature of having a digital board of tickets specially designed for SCRUM Projects.

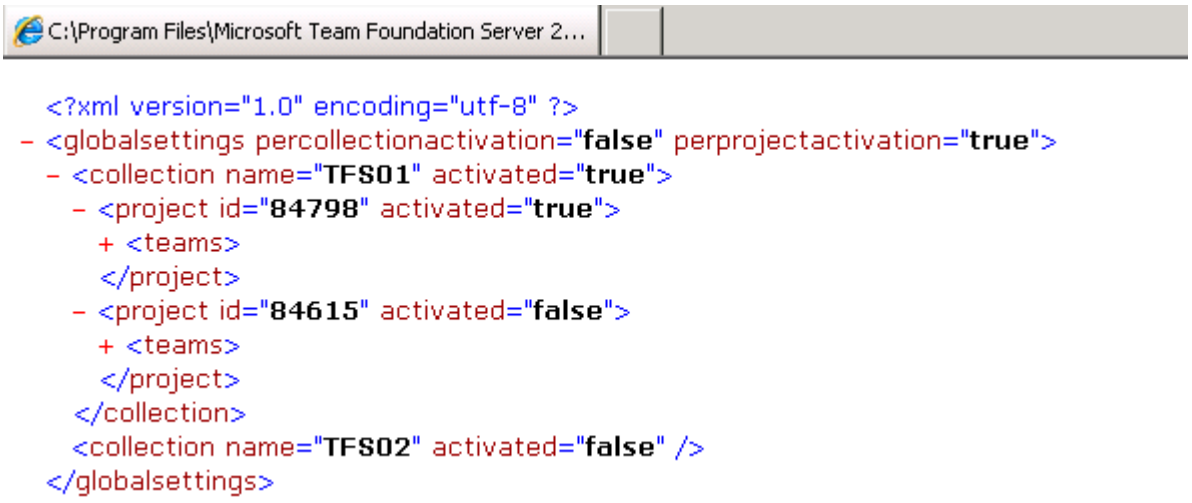
Hence, Urban Turtle is an application that can be used for SCRUM Projects which has got deadlines, timelines, bugs and requests, and other concepts of built in scrum methodologies.



The extended features include Planning Board and Task Board if that project collection to which the project belongs to, is enabled and activated in configuration of Urban Turtle on ETAS TFS Server. This can be found from the file called *global.settings* available under

C:\Program Files\Microsoft Team Foundation Server 2010\Application Tier\Web Access\Web\UrbanTurtle\configuration

The Activation of any Project under a Project Collection can be done by modifying the *general.settings* file available on the folder location in Microsoft Team Foundation Server.



If you look into the above file, then you can see the project id's that has been activated to *true*. Basically, this would be the project which shall be visible and activated in the Urban Turtle Web Access page.

The project id's of all the projects under Project Collection is shown in the screenshot below. If you want to activate a project having 750 as id under a project collection, then you need to modify the file by putting in these lines of code in the file.

```
<projects id = 750 activated = “true”>
        <teams>
        </teams>
</projects>
```

Collection name : TFS01

Project ID	Project name
750	010Meetings
83779	Company.EHI
85691	Company.TEF
83724	Dev.3rdParty
84605	Dev.ASCET SCM MSSCCI
84712	Dev.BuildSystem
83739	Dev.Components
83771	Dev.Components.General
83747	Dev.Components.Logger
83763	Dev.Components.Tools
84792	Dev.FWBuild
83755	Dev.Globals
84615	Dev.ITInternal
83730	Dev.MappingControl
84632	Dev.MDF
84756	Dev.NGDS
84798	Dev.SCRUM
84641	Dev.SystemTest
84626	Dev.VCI
7	DTB
84976	ES910FWBuild
384	LiMa
634	PSW.Components
224	SoftwareDevInfrastructure
400	SWEPFramework
57	TFSEvaluationProject

So, you will be able to see the Planning Board and Task Board as an extra feature to the project code 750 in the Web Access of Team Foundation Server.

Through the Urban Turtle, SCRUM Projects can work more effectively with much enhanced processes for carrying out their project related activities.

Advantage

- It allows the project managers to see the status of tickets being assigned to developers, testers and other team members.
- Project Managers can also create new sprints or releases and effectively manage the enhancements, bugs and other work item types.

6 ETAS Server Side Setup

Infrastructure for Servers at ETAS for TFS is mentioned in Section 4. Refer to the server information and other details.

The ETAS TFS Server is a combination of an app tier and data tier. The App tier is TFS Service running as an application/service. The data tier is the database engine that has all the data to run the TFS App Tier.

6.1 Application Tier

The ETAS TFS Application Tier has the following services installed

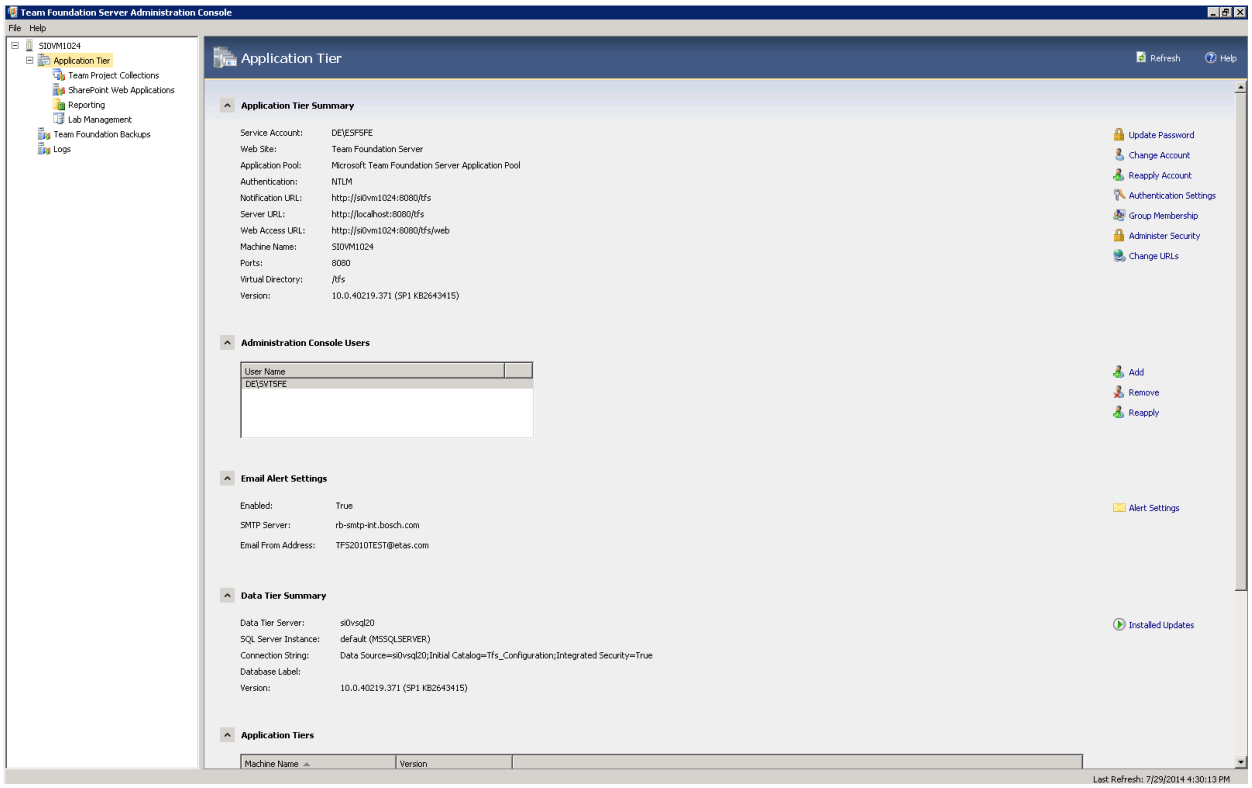
- Team Foundation Server Console
- Internet Information Services Manager (IIS)
- SQL Server Management Studio
- Build Server (Build Controller & Build Agent)

6.1.1 Team Foundation Server Console

Currently, TFS 2010 v 10.0.40219.371 is installed as a component of application tier.
The Team Foundation Server Console has

- Application Tier
  - Team Project Collections
  - SharePoint Web Applications
  - Reporting
  - Lab Management
- Team Foundation Backups
- Logs

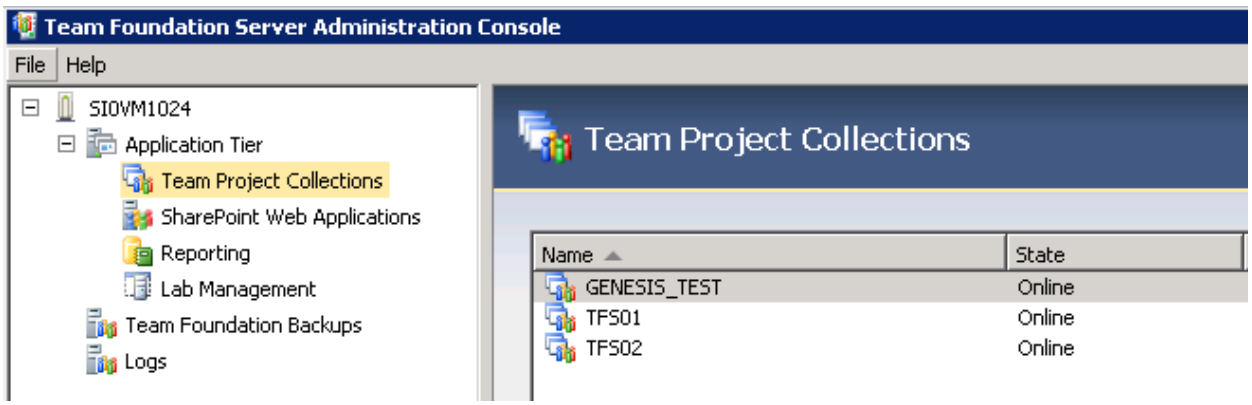
Below is a screenshot of the Team Foundation Server Console



TFS Console is running with service user DE\TRV9FE in Production System. Refer to other details in the above screenshot. Below is the table with service account users in all system.

Environment	Service User
Production System	DE\TRV9FE
Quality/Test System	DE\ESF5FE

We have TFS01, TFS02, and GENESIS_TEST as the Project Collections of TFS Server.



Currently, SharePoint Web applications and Lab Management are not configured in ETAS TFS Server.

Team Foundation Backups and Logs are also available as part of ETAS TFS Console. Logs can be viewed directly from the console without directly getting into the folder.

6.1.2 SQL Server Management Studio

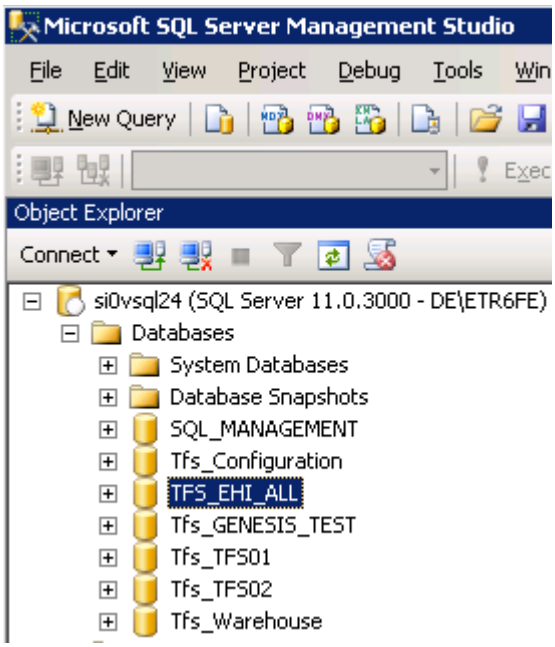
SQL Server Management Studio serves the purpose of connecting to SQL Server. SQL Server basically is the data tier.

The service account used in the connecting to SQL Server Studio is shown in the table below.

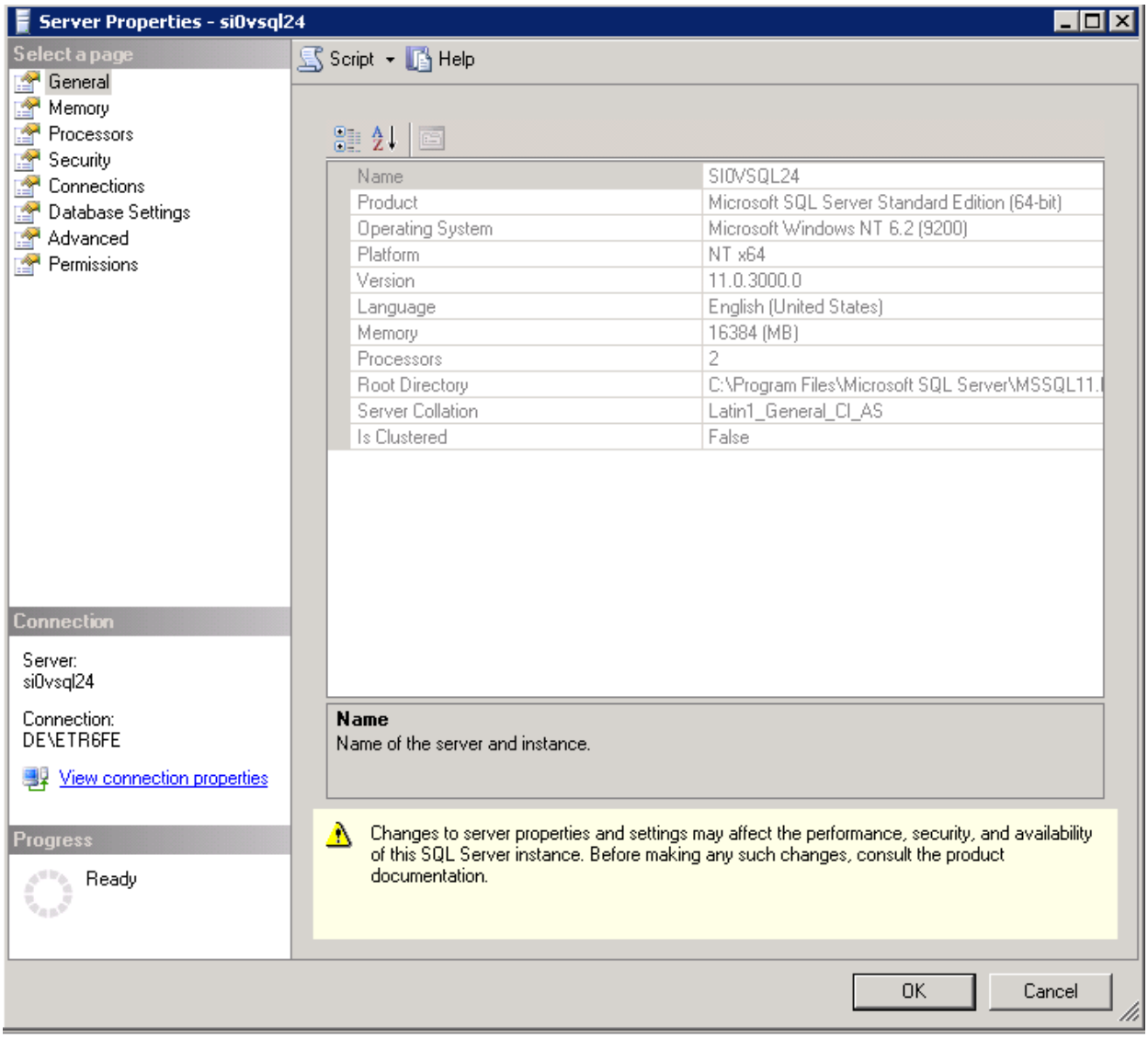
Environment	Service User
Production	DE\ETR6FE
Development	DE\SVT5FE

Currently, SQL Server 2008 R2 is used with Management Studio as the client interface to connect to the Server. DE\ETR6FE is being used as a service user to connect to SQL Server(P System).

As seen in the above screenshot, Mode of Authentication is Windows Authentication.



Right-Clicking on the Properties Tab of the database would give you a window something like below.



The window displays all the information like Product Name, Operation System, Platform, Version, Language and other details.

The value of Server Collation in above window is Latin1_General_CI_AS. Here CI stands for Case Insensitive and AS stands for Accent Sensitive. Also, Server is not clustered.

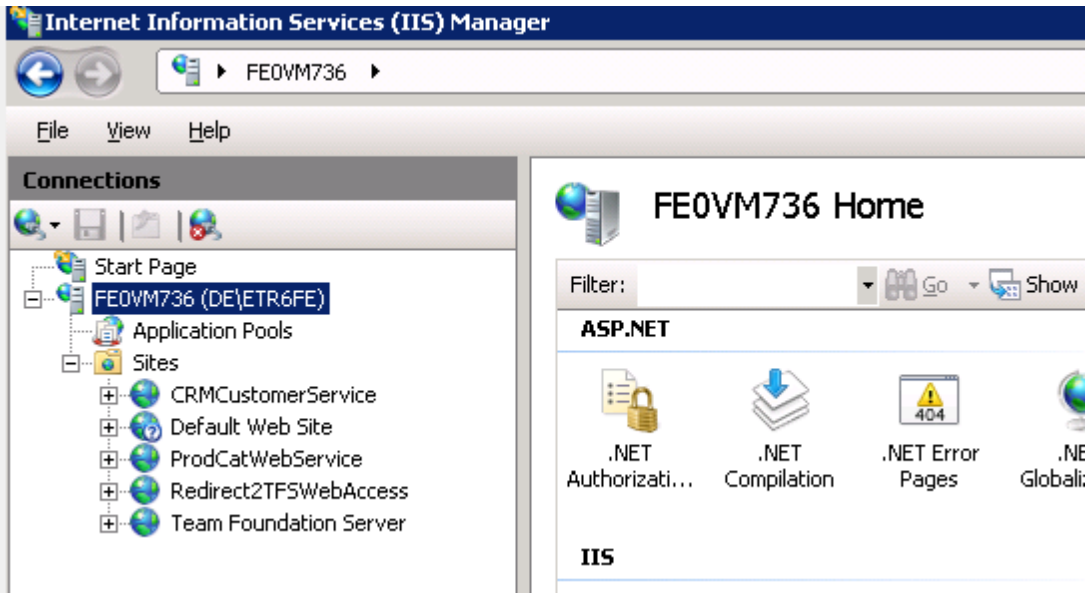
Now, as the Object Explorer clearly shows, there are databases under database instance and many tables under the database. Some very important databases are *TFS_CONFIGURATION, TFS_EHI_ALL, TFS_TFS01, and TFS_TFS02*

All the Project Collections created as part of TFS Console will have their dedicated databases.

6.1.3 Internet Information Services

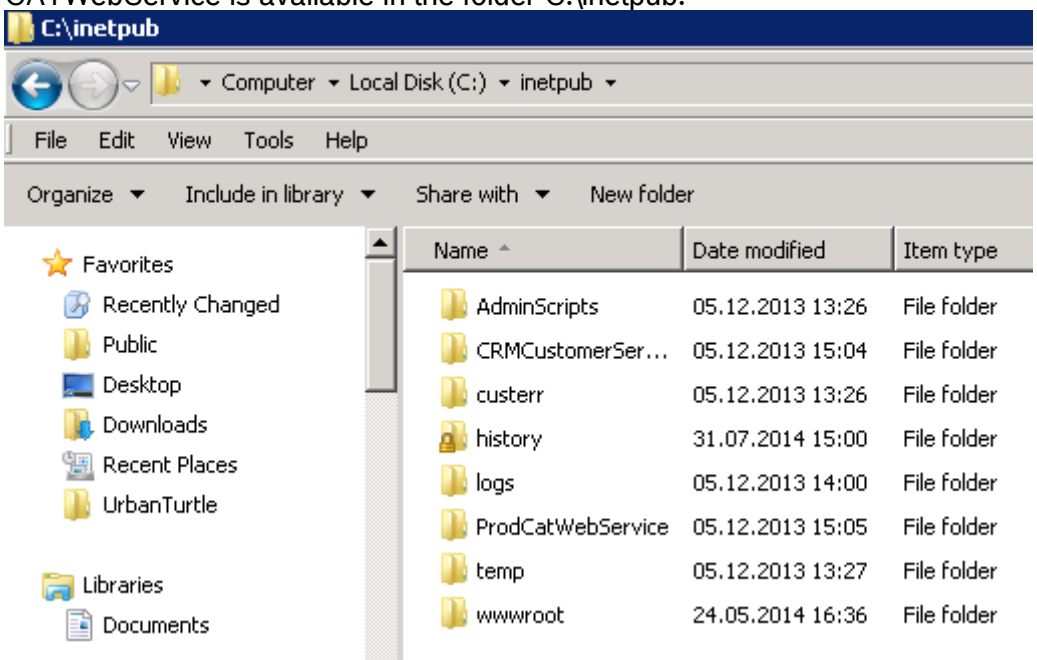
Internet Information Services Manager is also required to configure some components of ETAS TFS Server like Web Access, Web Services, and other web sites related configurations.

C:\inetpub serves the folder for IIS Services to host the websites.



As shown in the above screenshot, we have Team Foundation Server running as web site along with other components like CRMCustomerService, ProdCATWebService.

The Installation and Configuration for both the services i.e. CRMCustomerService and Prod-CATWebService is available in the folder C:\inetpub.



Basically, CRMCustomerService and ProdCATWebService are the services running in the IIS Services Manager to host or display the list of all the customers and products available in the database of the data tier to the front end.

Database Details:

The table below shows the details of the database and their tables which are used in IIS Services to host or display the data on to the front end.

Item	Database Name	Database Table Name
Customers list	[TFS_EHI_ALL]	[dbo].[SAPCRM_CustomerContact], [dbo].[SAPCRM_CustomerContactTemp]
Products List	[TFS_EHI_ALL]	[dbo].[SAP_ProductFamily], [SAP_ProductCatalog]

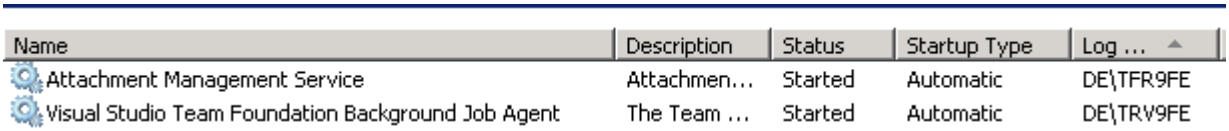
In case of Customers list, we have a main table and a temp table. Reason for this is that the service checks for customer data between the Temp and Main table, verify and validate before showing it up on to the front end.

Redirect2TFSWebAccess

In addition to above services, we also have **Redirect2TFSWebAccess** which takes care of redirecting the OLD URL to NEW URL, so users can still be able to access OLD URL.
For example, it redirects the browser http://fe20138s.ecn.etasgroup.com:8080/tfs/web/ to current one in the production without the need of putting the new URL.

6.1.4 Services

Visual Team Foundation Service Job Agent and Attachment Management Service are the only two services related to Team Foundation Server is running as a service in the application tier of Team Foundation Server.



Name	Description	Status	Startup Type	Log ...
Attachment Management Service	Attachmen...	Started	Automatic	DE\TFR9FE
Visual Studio Team Foundation Background Job Agent	The Team ...	Started	Automatic	DE\TRV9FE

For more details on Attachment Management Service, Please refer to Section 5.1.2

Team Foundation Job Agent is running as a service that runs both access pool and web access pool. It can be stopped/started/restarted using the services. This process is very well available under Task Manager Window.

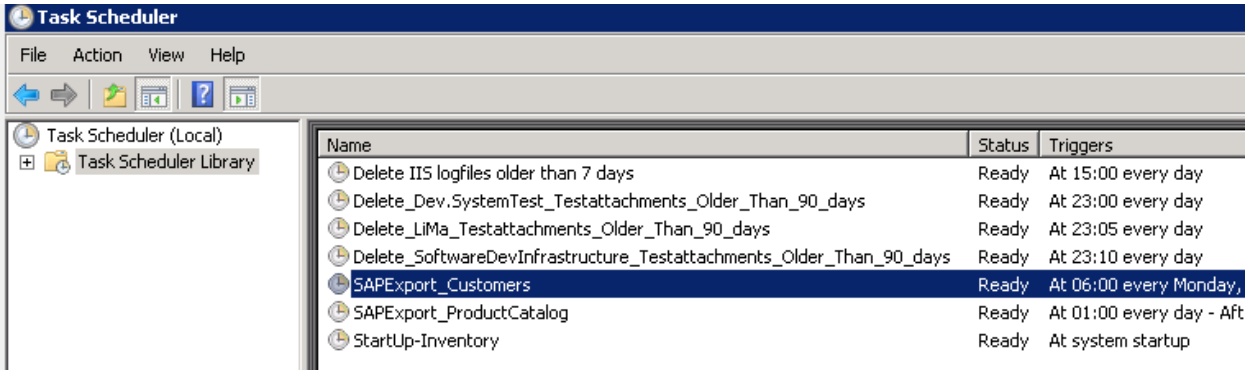
All the processes that is running either as a service or a process uses DE\TRV9FE as clearly shown in the Task manager screenshot.

de.ETAS.TfsAttachmentMgmt.WS...	TFR9FE	00	24.488 K	de.ETAS.TfsAttachmentMgmt.WSServiceh
w3wp.exe	TRV9FE	00	1.090.08...	IIS Worker Process
conhost.exe	TRV9FE	00	1.540 K	Console Window Host
TFSJobAgent.exe	TRV9FE	00	318.512 K	Visual Studio Team Foundation Background
w3wp.exe	TRV9FE	00	682.156 K	IIS Worker Process
w3wp.exe	TRV9FE	00	146.260 K	IIS Worker Process
SAP2TFSProductCatalogProvider....	TRV9FE	00	20.040 K	SAP2TFSProductCatalogProvider
taskeng.exe	TRV9FE	00	1.736 K	Task Scheduler Engine

6.1.5 Task Scheduler

Processes and Jobs are scheduled out of business and specific reasons to ensure smooth functioning of TFS Server like Log Files Deletion, Attachments Deletion.

The screenshot below shows all the jobs running under Task Scheduler.



- **SAPExport_Customers**: This process is used in calling SAP through web services to updating the database in the data tier with updated list of customer details and information. This process is scheduled to run for every 15 mins every day.
- **SAPExport_ProductCatalog**: This process is used in calling SAP through web services to update the database available in the data tier with updated list of products available from ETAS to the customer. This process is scheduled to run for every 1 hour in duration of a single day.

SAPExport_Customers uses the below action to run as a job on a regular basis.

Action	Details
Start a program	"C:\Program Files\ETAS GmbH\SAP2TFSProviderCustomer\SAP2TFSProductCatalogProvider.exe" /getcustomers:7

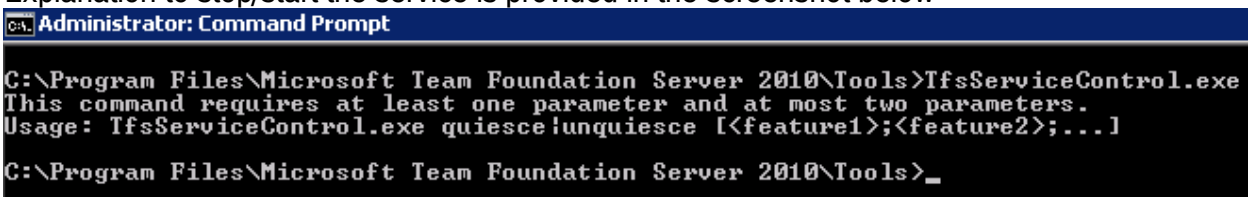
SAPExport_ProductCatalog uses the below action to run as a job on a regular basis.

Action	Details
Start a program	"C:\Program Files\ETAS GmbH\SAP2TFSProviderCustomer\SAP2TFSProductCatalogProvider.exe"

6.2 Stop/Start/Restart of TFS Server

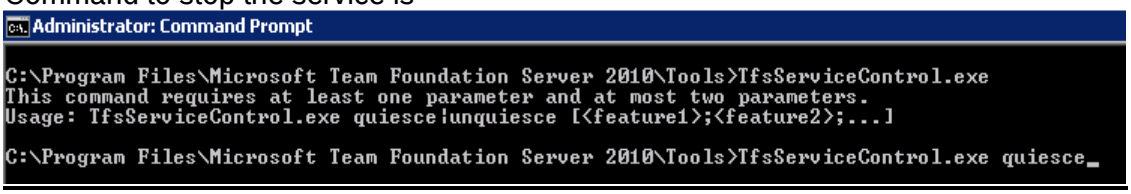
Team Foundation Server running in the Application Tier should only be stopped started or re-started during critical situations and should be avoided for any testing.

Explanation to stop/start the service is provided in the screenshot below

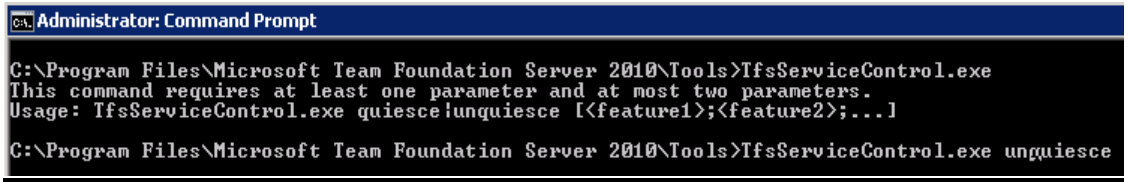


Short description

- Command to stop the service is

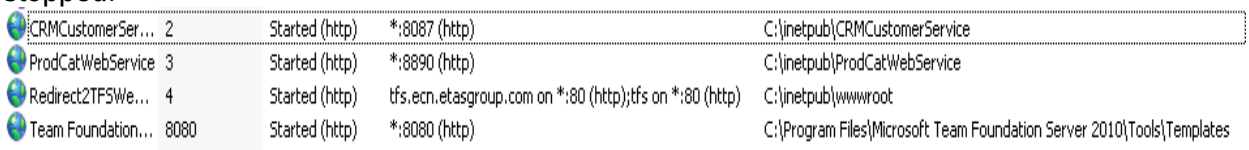


- Command to start the service is



If the service is stopped, then you can refer to IIS Services to check the status of all the server components running with in the Application Tier of Team Foundation Server.

Here, all services are running in below picture, Status would say Stopped if application is stopped.

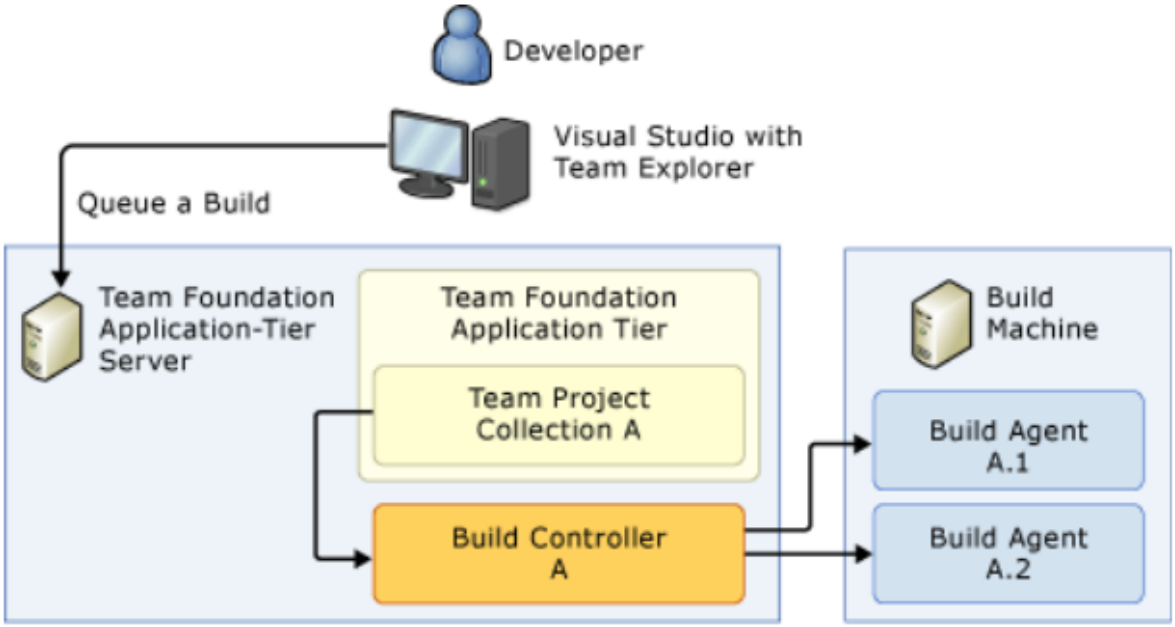


6.3 Build Server

The Build server is also a component of TFS Server. Build System is generally used to compile the source code and release different versions of software until it is fully built.

Build System generally contains a Build Controller with multiple Build Agents configured. Build Controller is like a master which assigns the job to Build Agents that act like a slave.

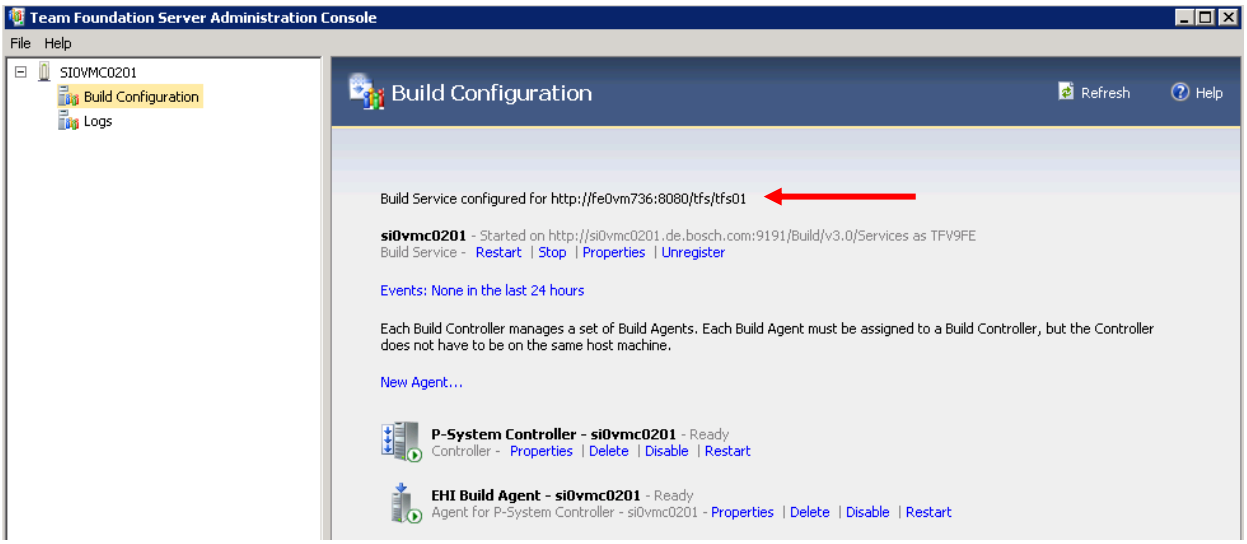
A Simple Build System



As you can see from the above screenshot,

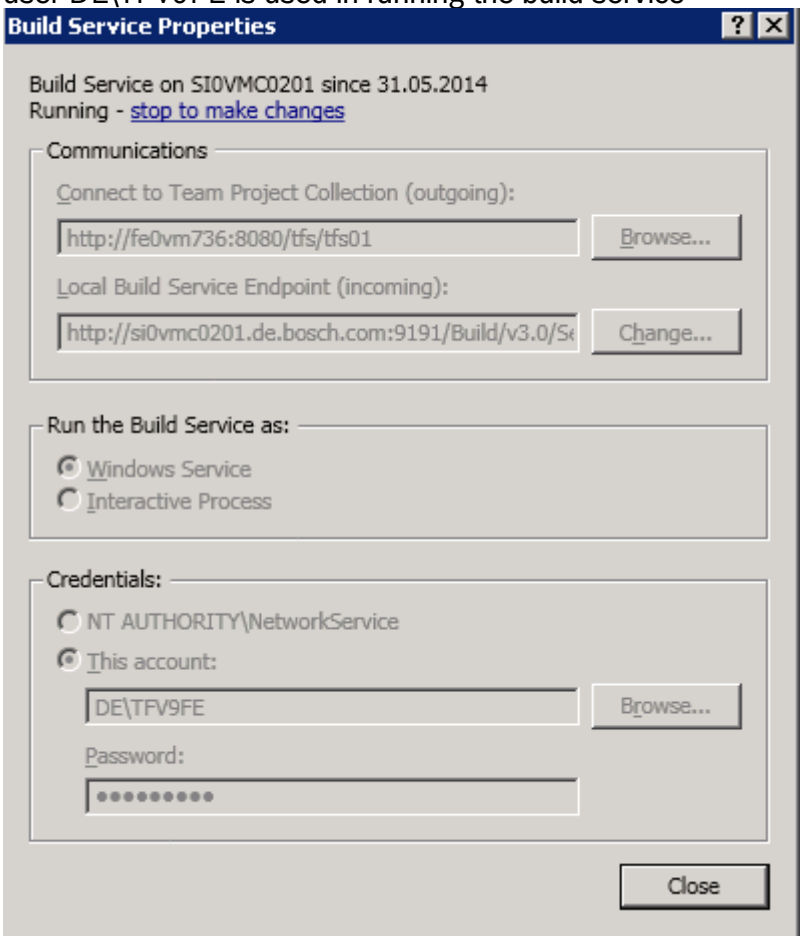
- Developer on the Visual Studio queues a build with respect to the source code to the Application Tier Server
- The Application Tier server founds the build controller linked to it. (Every Application Tier has a single build controller)
- Multiple Build Agents are linked to a single build controller
- Build Agents processes the queue based on its availability and builds the source code based on the build definition file.

At ETAS Build System used by Application Tier of TFS Server, SI0VMC0201 acts a build controller with a single agent configured on it.



From the above screenshot, it is clear that Build Service is configured for the Application Tier server.

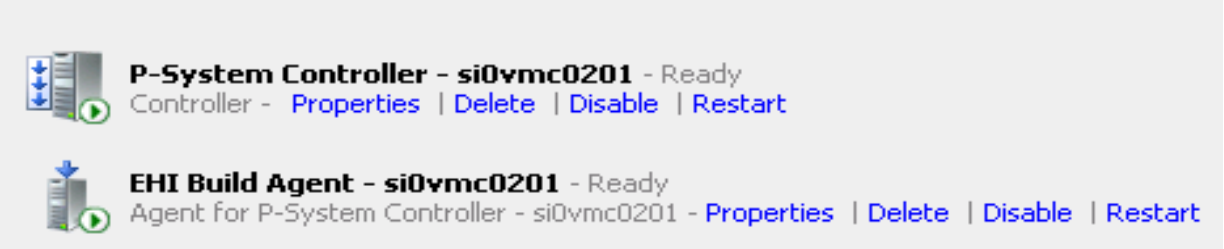
Clicking on the properties option available for the Build Service, We understand that service user DE\TFV9FE is used in running the build service



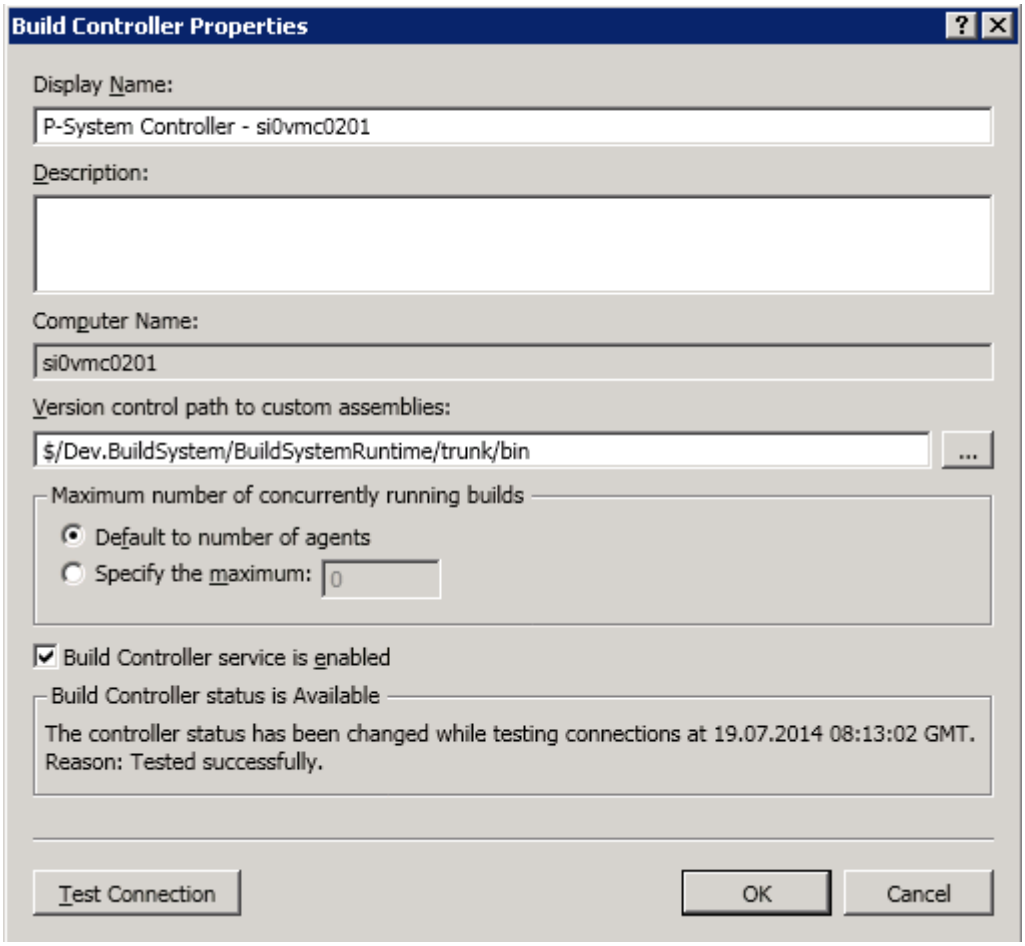
It is very well connected to TFS Application Tier FE0VM736 and is running on port on 9191.

As we also understand from the above discussion that Each Build Controller manages a set of Build Agents. Each Build Agent must be assigned to a Build Controller, but the controller does not have to be on the same host machine.

At ETAS ICO, we have a single build controller with a single build agent.

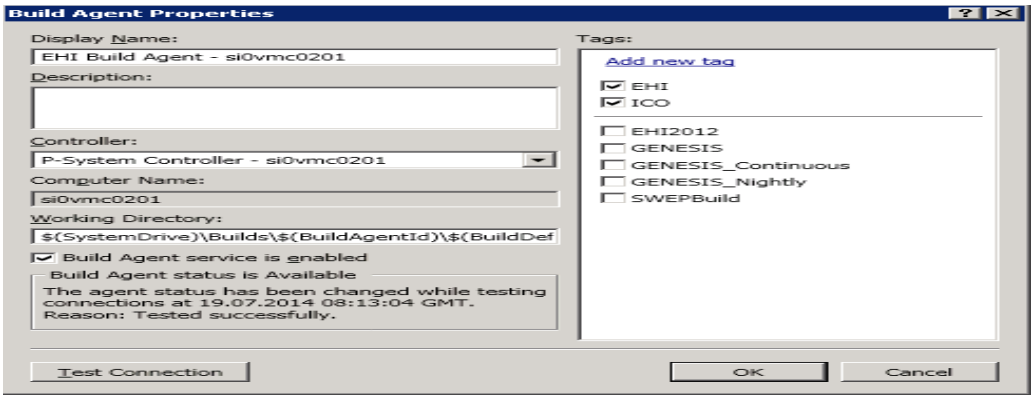


The name of the Build Controller is *P-System Controller - si0vmc0201*



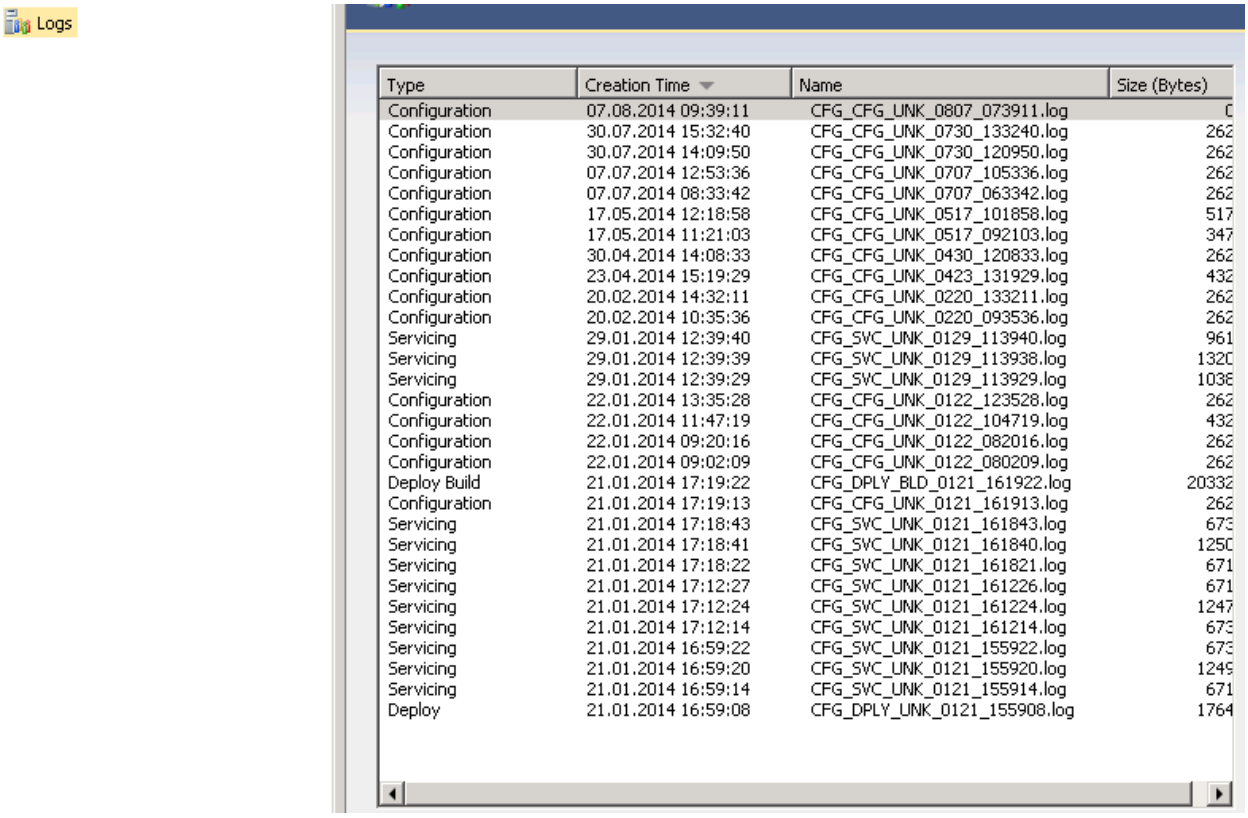
Version control path to custom assemblies is *$/Dev.BuildSystem/BuildSystemRuntime/trunk/bin*
As we can see that, we have several options like Properties, Delete, Disable and Restart.

As of now, a single build agent is managed by build controller. *EHI Build Agent – si0vmc0201* is the name of build agent. Properties option will provide other details as shown below.



This build agent can be configured to be managed by a build controller by selecting the name of the controller in the above screenshot. The Working directory for the build agent can also be out down to enable the build agent to build the source code in the working directory.
In our case, the source code under Company.EHI is used in building and hence we have the name called EHI available as a tag under Build Agent.

The Logs concerning the build system is available in the local directory of the machine where build controller is hosted.

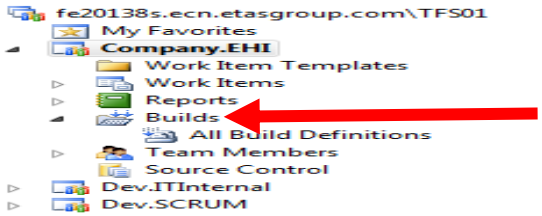


Type	Creation Time	Name	Size (Bytes)
Configuration	07.08.2014 09:39:11	CFG_CFG_UNK_0807_073911.log	0
Configuration	30.07.2014 15:32:40	CFG_CFG_UNK_0730_133240.log	262
Configuration	30.07.2014 14:09:50	CFG_CFG_UNK_0730_120950.log	262
Configuration	07.07.2014 12:53:36	CFG_CFG_UNK_0707_105336.log	262
Configuration	07.07.2014 08:33:42	CFG_CFG_UNK_0707_063342.log	262
Configuration	17.05.2014 12:18:58	CFG_CFG_UNK_0517_101858.log	517
Configuration	17.05.2014 11:21:03	CFG_CFG_UNK_0517_092103.log	347
Configuration	30.04.2014 14:08:33	CFG_CFG_UNK_0430_120833.log	262
Configuration	23.04.2014 15:19:29	CFG_CFG_UNK_0423_131929.log	432
Configuration	20.02.2014 14:32:11	CFG_CFG_UNK_0220_133211.log	262
Configuration	20.02.2014 10:35:36	CFG_CFG_UNK_0220_093536.log	262
Servicing	29.01.2014 12:39:40	CFG_SVC_UNK_0129_113940.log	961
Servicing	29.01.2014 12:39:39	CFG_SVC_UNK_0129_113938.log	1320
Servicing	29.01.2014 12:39:29	CFG_SVC_UNK_0129_113929.log	1038
Configuration	22.01.2014 13:35:28	CFG_CFG_UNK_0122_123528.log	262
Configuration	22.01.2014 11:47:19	CFG_CFG_UNK_0122_104719.log	432
Configuration	22.01.2014 09:20:16	CFG_CFG_UNK_0122_082016.log	262
Configuration	22.01.2014 09:02:09	CFG_CFG_UNK_0122_080209.log	262
Deploy Build	21.01.2014 17:19:22	CFG_DPLY_BLD_0121_161922.log	20332
Configuration	21.01.2014 17:19:13	CFG_CFG_UNK_0121_161913.log	262
Servicing	21.01.2014 17:18:43	CFG_SVC_UNK_0121_161843.log	673
Servicing	21.01.2014 17:18:41	CFG_SVC_UNK_0121_161840.log	1250
Servicing	21.01.2014 17:18:22	CFG_SVC_UNK_0121_161821.log	671
Servicing	21.01.2014 17:12:27	CFG_SVC_UNK_0121_161226.log	671
Servicing	21.01.2014 17:12:24	CFG_SVC_UNK_0121_161224.log	1247
Servicing	21.01.2014 17:12:14	CFG_SVC_UNK_0121_161214.log	673
Servicing	21.01.2014 16:59:22	CFG_SVC_UNK_0121_155922.log	673
Servicing	21.01.2014 16:59:20	CFG_SVC_UNK_0121_155920.log	1249
Servicing	21.01.2014 16:59:14	CFG_SVC_UNK_0121_155914.log	671
Deploy	21.01.2014 16:59:08	CFG_DPLY_UNK_0121_155908.log	1764

You can refer to logs to do further analysis and can use it for investigation purposed related to any build failures.

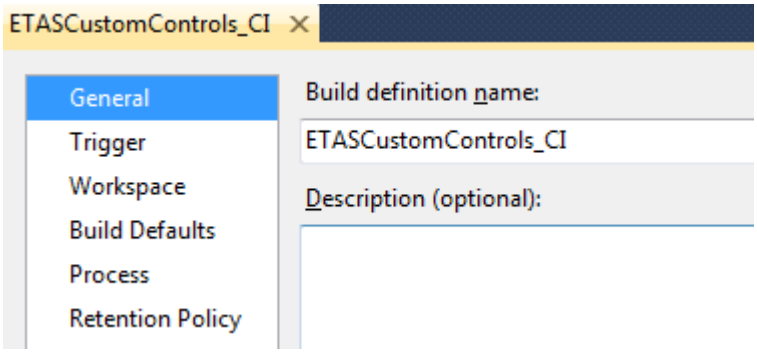
6.3.1 Build Configuration at client side

Build configuration at client side by creating a proper build definition. Please refer to screenshot below.



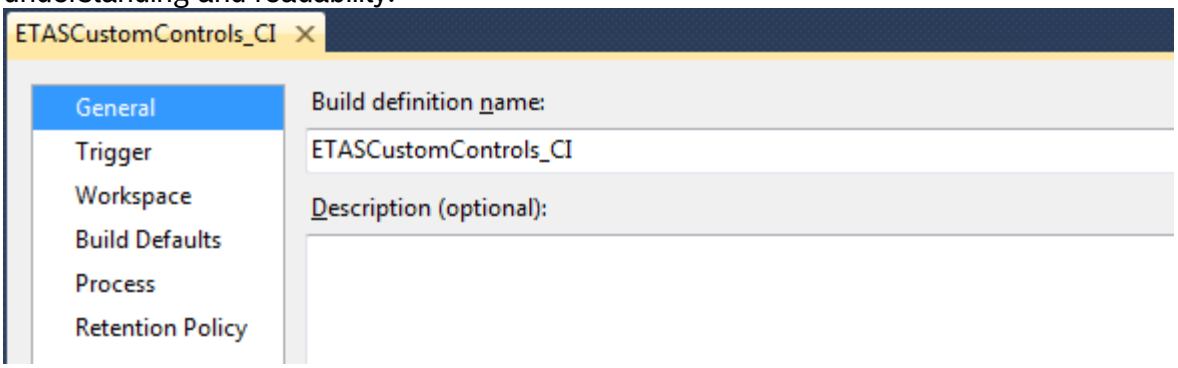
If you expand the Build option shown above, you can see option "All Build Definitions" which would have all definitions configured for that particular project's source control.

At Project Dev.ITInternal, We have a build definition called "*ETASCustomControls_CI*".
On Right-clicking to find Edit Build Definition, We would see the definition and configurations defined for that build.

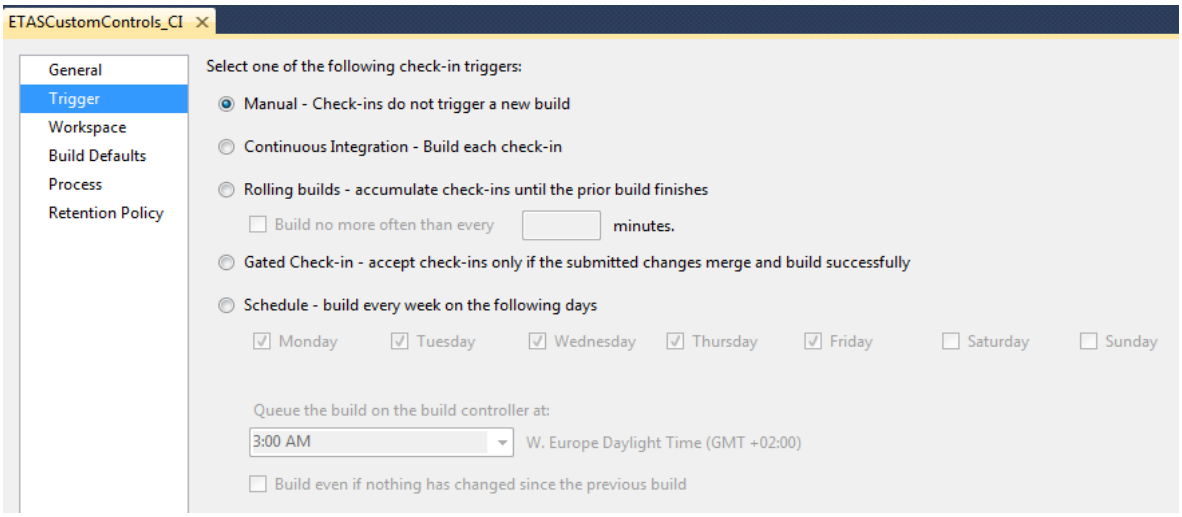


There are 6 major definitions, i.e. General, Trigger, Workspace, Build Defaults, Process, and Retention Policy

- General: Here, name of the build definition is provided with proper name pattern for eas understanding and readability.

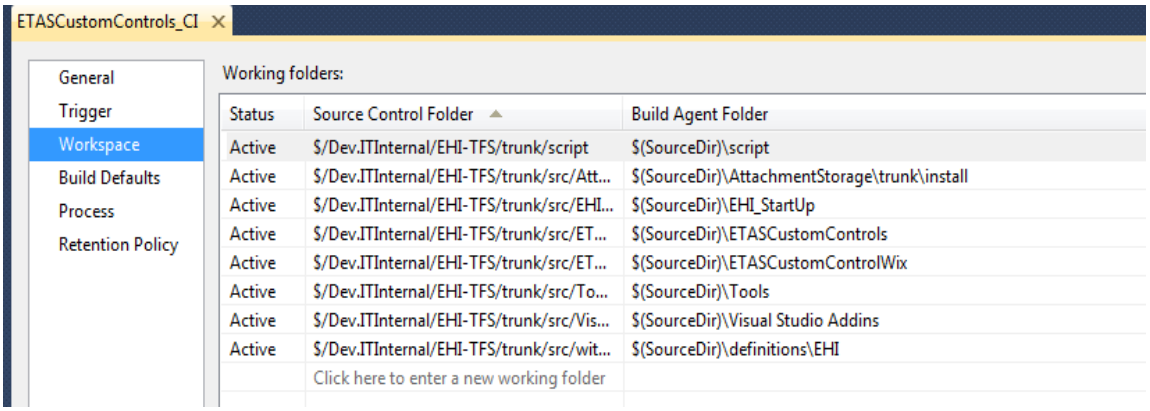


- Trigger



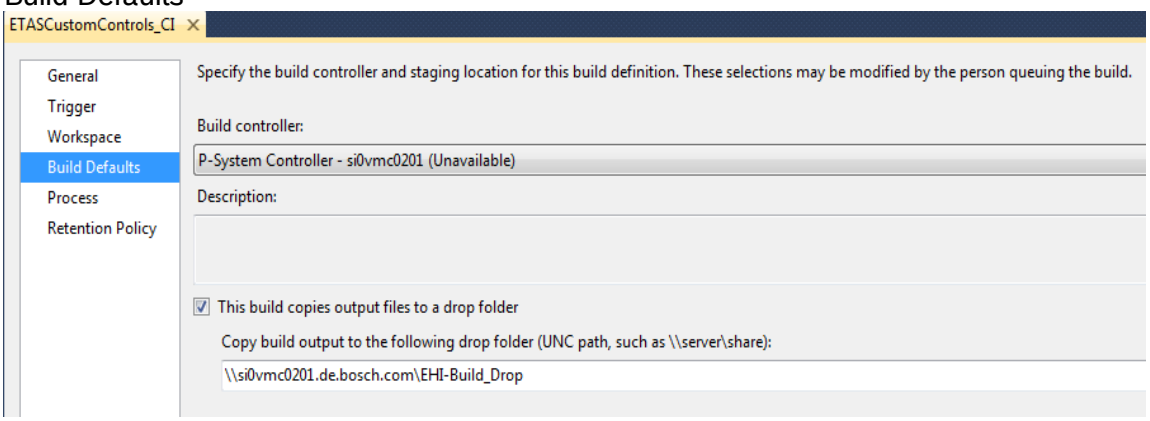
Here, we can provide the details about the initiation of the build that would trigger on some action. Above some options are available for the trigger to take place.

- Workspace: Here, we provide the details of the source code control that would required to be built.



Please note that the source control folder refers to all the source code available under the source control of that particular project.
Build Agent folder refers to $(SourceDir) → C:\Builds\208\Dev.ITInternal\ETASCustomControls_CI available on the build server. The above value depends upon the build number.

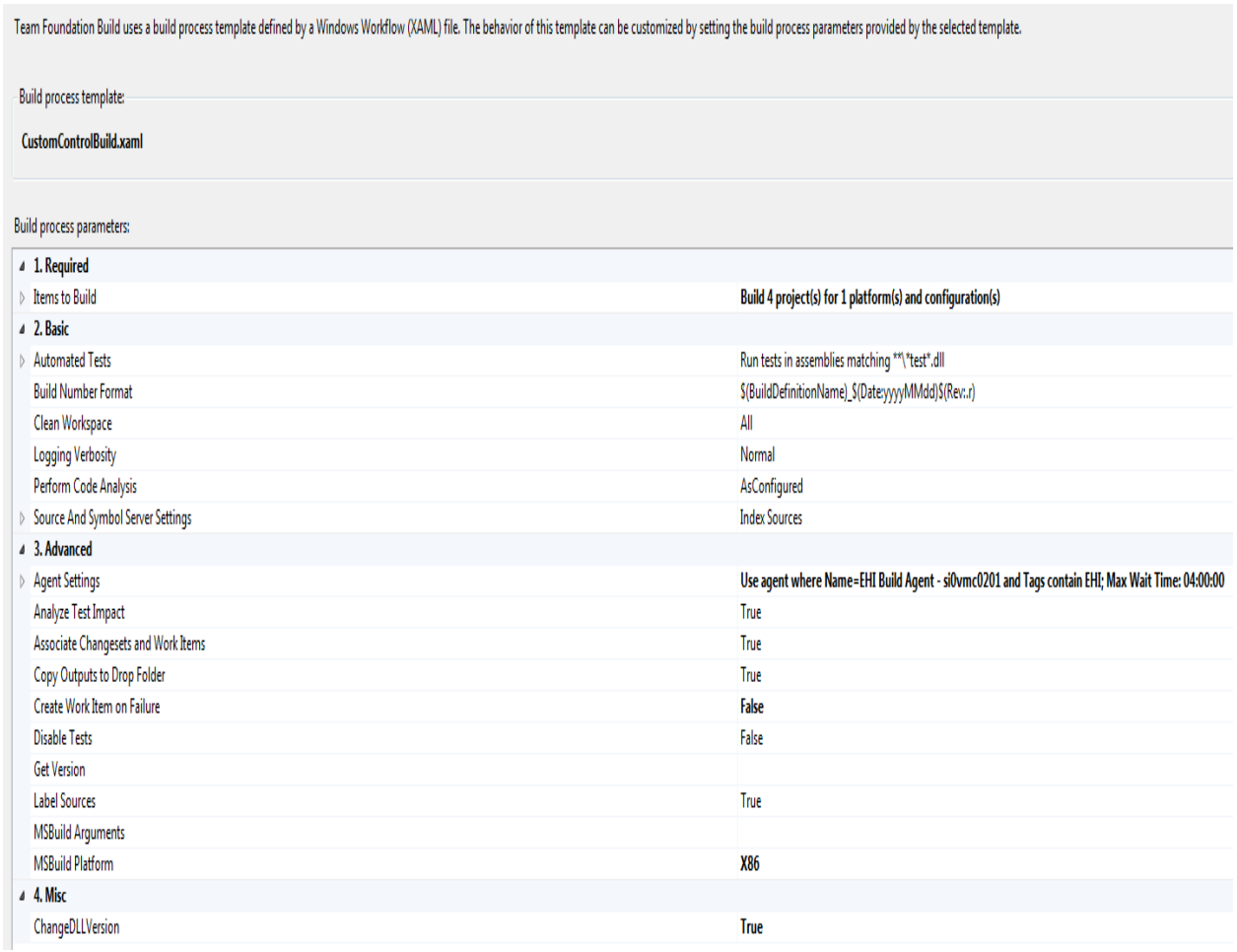
- Build Defaults



Here, we define the build controller.

- Process

Here, we can define the build agents and other necessary details required for build.
Also, if we expand the advanced option available in the above screenshot, then it would give the name of the build agents that is configured to run the build.
See below Screenshot.



Build agent named EHI Build Agent – si0vmc0201 is configured for the build which has a wait time of around 4 hours.

Please note that, many development teams have their own controllers and build agents configured to be able to build their source code out of their own source code projects.

7 CONCLUSION

Thanks for going through the TFS Admin Tutorial designed and developed during ETAS TFS Transition Period from 28-07-2014 to 08-08-2014