

WorkItems migration to VSTS. Very brief notes on tools.

Version 26 May 2017

Contents

WorkItems migration to VSTS. Very brief notes on tools.	1
Summary	2
1/ Create Teams	2
2/ Create Template with From Id.....	2
3/ Migrate iterations and area paths.	3
Prerequisites.....	3
Steps.....	3
Use the Export functionality to generate a node file.....	3
4/ Migrate Team paths dates and settings.	5
5/ Migrate workitems.	5
Prerequisites	5
Steps.....	5
6/ Migrate Test Plans.....	7
Prerequisites	7
Steps.....	7
Warning!	8

Summary

- Workitems Migration is in two parts:
 - Workitems
 - Test Plans with associated artefacts.

Both based on query. Note that for test plans the Test Plan workitem/s are what drive the migration.

- Teams should exist prior to migration on the VSTS target.
 - They should be the same name as they are on the TFS on premise.
- Migration template with **From Id** needs to have been set up and applied.
- Teams should be set up in advance so that team members can test their access.
 - This primes the VSTS with the new user which means the workitem migration will be able to use that user for the Assigned To and other user related fields

There are three tools:

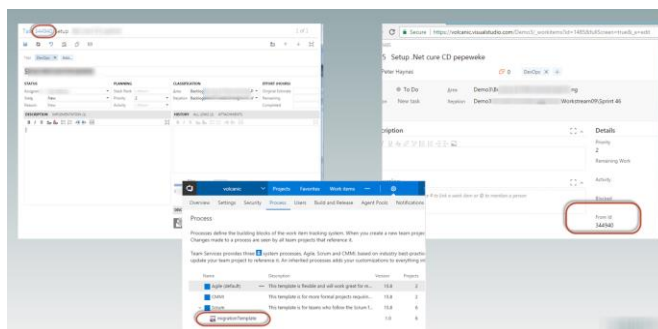
- **AreaImportExportTool.**
 - To migrate iterations and areas.
 - Based on the tool written by Neno Loje.
 - <http://blogs.msmvps.com/vstsblog/2010/07/08/updated-area-import-export-tool-for-tfs-2010/>
- **QueryRunImport.**
 - To migrate Workitems based on a query.
- **TestCasesPlans.**
 - To migrate Test Plans based on a query.

1/ Create Teams

Create the Team in the target VSTS. Assign the users and have the users check access.

2/ Create Template with From Id

See below an example of a “migrationTemplate” that has been created and used. All the workitems in that template have been given a “From Id”. When the Workitem and Test Plan migration tools create the workitem in the target VSTS system the “From Id” is populated with the Id of the source workitem. This allows the tool to be re-run. The tools check the “From Id” and if it matches the workitem being migrated the tool identifies it as existing and will not re-create it. There is scope to improve this area of the tools; in particular using a string for “From Id” rather than an integer. Using a string would allow the collection to be added to the id of the workitem and avoid the use of an offset for multiple collections migration.



Use the **AreaImportExportTool**.

Target Team exists with users assigned on the target.

The best approach is to run the queries that the team has produced for their migrations and eyeball the area and iteration paths and **only** upload the returned iteration and area paths. (Note for future development – functionality to get the iteration paths from the queries). This avoids an excessive number of irrelevant paths being uploaded to the target VSTS.

RippleRock - Area Import/Export Tool - DISCLAIMER Use this tool at your own risk! Test it out before running against a live system!

Source TFS:

Collection:

Team Project:

Target TFS:

Collection:

Team Project:

Choose items to im-/export

☒ Area structure

☒ Iteration structure

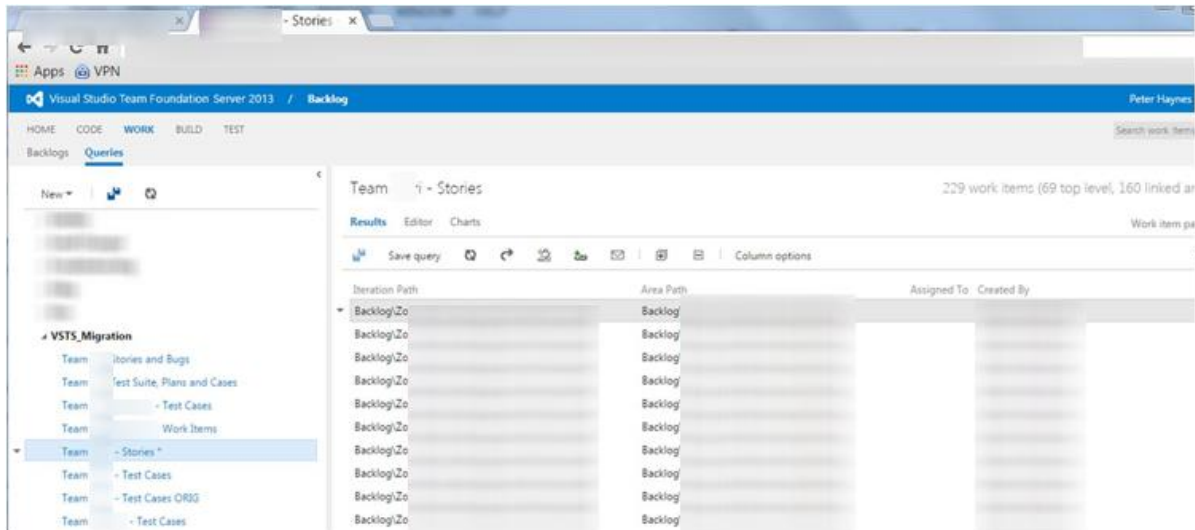
Teams paths migration

Team name in target must be same as source.

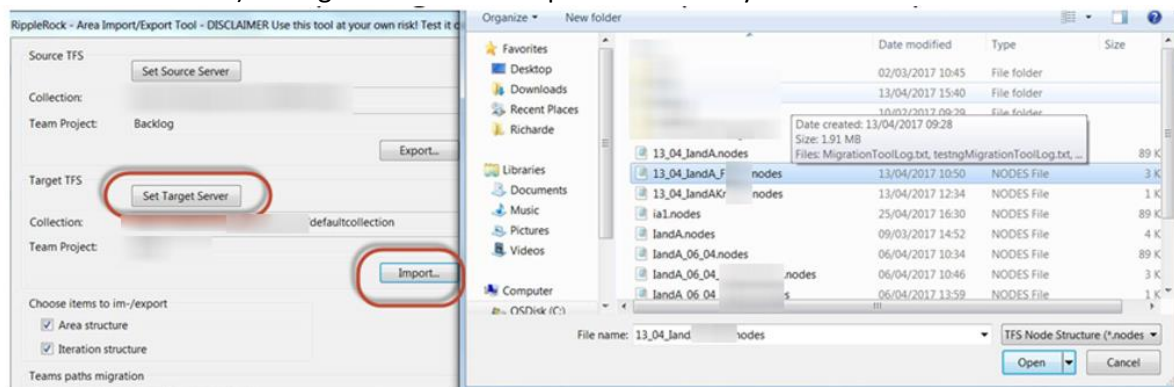
Set the source server.

[illegible]

Page 3 of 8



TBD - Update AreaImportExportTool to use query. (Note I've got by so far but would like to do this)
 Following edit of the exported paths upload the node file (I save to a different file name based on the name of the team) to target TFS. Use the Import functionality.



NOTE - Missing paths will cause the migration to fail.

4/ Migrate Team paths dates and settings.

Use the AreaImportExportTool Migrate Team Paths functionality,
Enter the Team name. Team names **must** match on source and target. (NB also enter a source and target TFS)

The screenshot shows the 'AreaImportExportTool' dialog box with the 'Migrate Team Paths' section active. The 'Source TFS' section has 'Collection' set to 'http://...:8080/tfs/' and 'Team Project' set to 'Backlog'. The 'Target TFS' section has 'Collection' set to 'http://...faultcollection' and 'Team Project' set to 'Backlog'. Under 'Choose items to im-/export', both 'Area structure' and 'Iteration structure' are checked. In the 'Teams paths migration' section, the text states 'Team name in target must be same as source.' and the 'Team' field is set to 'Team'. The 'Migrate Team Paths' button is highlighted with a red circle. Other buttons include 'Set Source Server', 'Set Target Server', 'Export...', 'Import...', and 'Close'.

TBD - Update AreaImportExportTool to allow migration to different team. (Note I've got by so far but would like to do this)

5/ Migrate workitems.

Use the QueryRunImport tool.

Prerequisites

Migration template with From Id has been created and applied.

Iteration and Area paths have already been migrated, see above.
Users exist (In Team) on the target with validated access, see above.

Steps

Enter Source and Target TFS and Team Project.

Enter Query (supplied by team being migrated) see below. Copy it exactly.



Only if migrating from a second TFS source like a pilot enter a value in Offset the from Id.

- Leave Override All blank.

Page 6 of 8

6/ Migrate Test Plans.

Prerequisites

Migration template with From Id has been created and applied.

Iteration and Area paths have been migrated
Users exist on the target with validated access

Steps

Enter Source and Target TFS with Team Project.

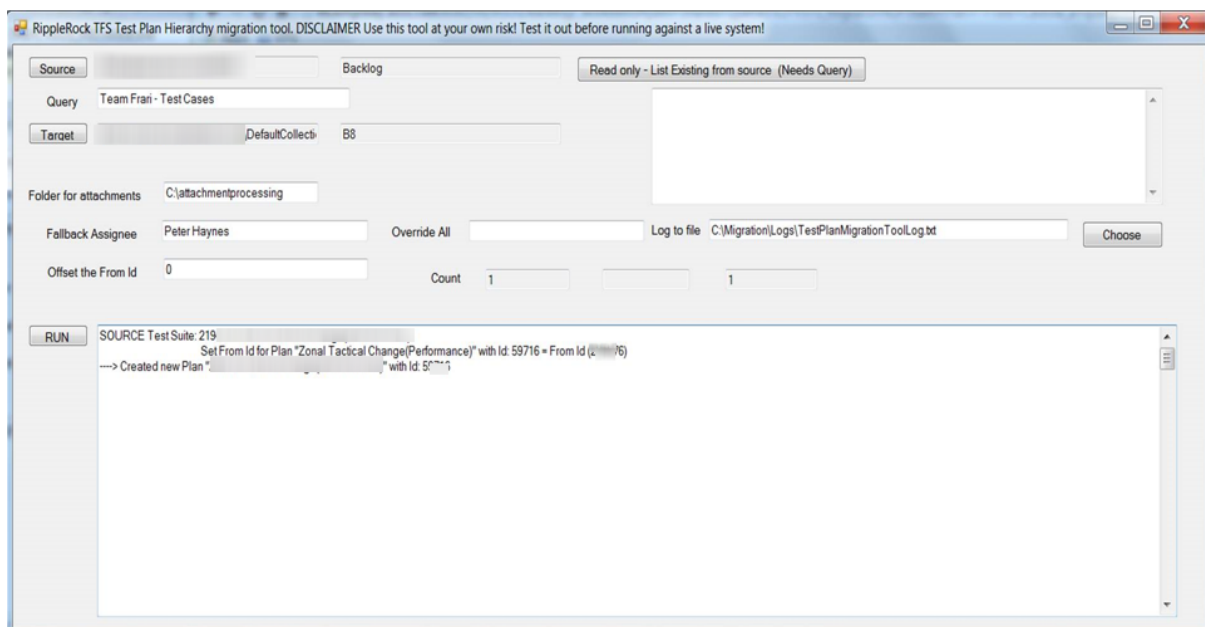
Enter Query.

Enter Fallback Assignee.

Only if migrating from a second TFS source like the pilot enter a value in Offset the from Id.

- This is quite rare. i.e. migrate from pilot.
- This is important and is to avoid a clash around the from id.
 - The pilot has same ids as the on premise TFS and they need to be distinguished.
 - This is the workaround.
- E.g For pilot VSTS use offset of 1000000 (keep a record so always use same)
- TBD – Use a string rather than an integer for the from field and pre-pend the id with the collection guid – code needs updating for this

Leave Override All blank.



That should it be set up and ready to run.

Following migration check using MTM to validate.

Warning!

*I suggest strongly to anyone using these tools for the first time that a few practice migrations are run against a **pilot** as it is important to get a feel for the tools and how they work.*

Create a brand new Team Project and then run through the steps to use the tools as above.

Run in Visual Studio in Debug mode to step through migration of the first few workitems. These tools are a very powerful “workbench” for migration rather than a set of ready to run migration tools and need to be used with understanding and caution!