**Background/Summary**: Team Foundation Server (TFS) has been continuing to grow as a popular choice for Configuration Management within the Public Sector Software Development market. While it is still very common for a Government development team to only leverage TFS for Source Control features initially, more and more customers are showing desire to have better reporting, visibility and traceability within their development processes and very slowly realizing that this can be provided by leveraging additional features of TFS. This slow realization is essentially what sparks the initial demand for higher-cost features of Visual Studio features that can complement a strong ALM strategy facilitated by TFS.

**The USPS Market Challenge**: As with many customers and industries, the US Public Sector development shops consider themselves “different”. In working with 1000’s of software development teams across all Government the PS-DPE sales team would agree with that assessment, however, the “different” is best defined as “immature” or “antiquated”. Thus, bringing solutions to this market that are very forward-thinking and cutting-edge are often intimidating and overkill for these teams. Solutions that provide simplicity, minimal ramp-time and easy-to-understand value is what these teams find most attractive. In the Commercial business, IT and software can be used as a strategic advantage over competitors. In the world of no-competition and shrinking budgets this market is “different” because their motivations and success criteria are different.

**Solution**: Because the core value props of TFS are directly in line with Government interests (i.e. low-cost, flexible, familiar), there is tremendous opportunity to accelerate the adoption/sales of TFS and related products. The USPS-DPE sales team strongly believes that an optimal way of achieving this is leverage the TFS customization capabilities and create a TFS demonstration that allows a Government customer to actually visualize how it could be used in their teams today. Furthermore, upon seeing this customized TFS in demonstrations, seminars, marketing campaigns, etc, the teams will be further compelled to take the next step to adoption/sale when learning that the customized version is available to them via an open source accelerator package; “GovDev for TFS Accelerator”. This accelerator will include a number of assets that will streamline the customization of TFS to align with the most commonly seen approaches to software development processes in Public Sector development shops today.

The *GovDev for TFS Accelerator* will provide 3 core elements in **phase 1** and all be installed via a package install process:

* + GovDev TFS Process Template
  + A set of custom reports that map to the GovDev TFS Process Template (displayed on SharePoint Portal)
  + Integrated Process Documentation
  + Enable key settings: Check-in (WI association & comments), “Nightly Build Dev” (unit test, CA, CC, IntlliTrace)

The *GovDev for TFS Accelerator* **phase 2** goals are:

* + Expand the install process to optionally attach a sample TFS project with sample data built leveraging the GovDev TFS Process Template
  + A set of pre-built supporting assets: Microsoft Project File, Microsoft Word File with TeamSpec Skin, etc.
  + A Visual Studio Start Page Gadget pointing to the [USPS DPE Blog](http://blogs.msdn.com/b/publicsector/).
  + Other useful utilities with examples leveraging the sample TFS data (i.e. [TestScribe](http://visualstudiogallery.msdn.microsoft.com/e79e4a0f-f670-47c2-9b8a-3b6f664bf4ae/))

**Solution Requirements:**

GovDev TFS Process Template: The Process template shall be a combination of the current CMMi template and the Agile template. This approach should be taken to achieve the commonly seen “Formal-Agile-Formal” or “Waterfall-Agile-Waterfall” or “W-agile” methodology that sets out to meet the formality of Waterfall based Project methodologies but still leverage iterative coding and testing cycles where possible.

Ultimately, we are customizing the CMMI template as it is the more formal (water-fall approach) and adding in support elements that allow Agile iterations. Process documentation could make sue of the template documentation downloads on Visual Studio Gallery to ensure a consistent look and feel (CMMI - <http://archive.msdn.microsoft.com/MSFCMMIGuidance> and Agile <http://archive.msdn.microsoft.com/MSFAgileGuidance> .

|  |  |
| --- | --- |
| **Work Item Type** | **Notes:** |
| Requirement | From CMMi. |
| Use Case | IJI? |
| Change Request | From CMMi. |
| Support Ticket | New Type. May want to use the Support Request WI type from the recent MSDN article: <http://msdn.microsoft.com/en-us/magazine/hh335060.aspx> |
| Bug | From CMMi. |
| Task | From CMMI or SCRUM |
| Impediment | New Type |

Test Case: While Test Case is a default Work Item, we want to direct the user to leverage this Work Item even if they are no using VS Test Pro. In this case, they user attached a Word Doc to represent the Test Script vs. using he manual test features of Test Pro. *(this will enable the process to work and seed demand for Test Pro)*

Reports: All relevant existing TFS reports from the CMMi and Agile templates should be included in the GovDev for TFS Accelerator. In addition these reports should also be created:

* [Traceability Matrix](http://en.wikipedia.org/wiki/Traceability_matrix) (with Drill Down): Report(s) that shows linkages from individual Requirements to Use Case(s) to ChangeSets to Tests to Bugs. Relevant items should be hyperlinks and drill down to individual items on click.
* Developer Churn: Report(s) that show individual productivity views. (i.e. Per Developer… lines of code, # of unit tests, # of Check-ins, Bug Closeout, etc)
* *\*See Appendix A for more detail on Reports*

Other Components:

The GovDev Accelerateor for TFS should be implemented by the customer using an Installer package to simplify the set up. During this install the following additional changes should be set on TFS:

* Check-in Policy: The “Comments” and “Work Item Association” rules should be turned on.
* Build Profiles: 2 Build Profiles should be added – “Nightly” and “Continuous Integration”

Also during the install processes, relevant content should be added to a folder directory on the machine. Suggestions for that should include:

* Documentation Folder: Any/ All documentation relative to the specifics of the Customized Process Template, reports and default settings. *For example, why the Check-in Policy is “on” and what it means to turn it off relevant to the reports*.
* MSProject Folder: A sample MS Project file that is pre-populated and pre-mapped to work with the GovDev Template
* Requirements Mgt Folder: Include a pre-built/configured skin and trial software info/link for Team Spec.
* Other Suggestions?

**Distribution:** Upon completion of the GovDev Process Accelerator, it shall be made available on codeplex for all anyone to download. The vendor who works with Microsoft to build this asset will not retain any rights but has to right to add their company name, logo and contact information in the Process Documentation that encourages the end-user to contact them for consultation or training.

Appendix A:

Shows mockups of desired reports. These are to be used as guidelines but partner is free to tweak/add as necessary using industry experience. Also, not every report is defined here. Partner should use judgment to decide if additional reports need to be created in support of the core reports. For example, a desirable drill-down on a report may need to renderer a secondary report to show the appropriate/expected data.

\*TIF files can be downloaded from [here](https://skydrive.live.com/?cid=f59652f79267c2d5&sc=documents&id=F59652F79267C2D5%21454).





