This is the final chapter in our assessment and metrics series. We will cover some metrics and reports for release management, including the build process.

Metrics

Much of the metrics we can get from TFS are presented as reports. Using these figures as KPIs we can easily compare our progress when it comes to build quality.

A quick look at Information Technology Infrastructure Library (ITIL) (http://www.itilnews.com/ITIL_v3_Suggested_Release_and_Deployment_KPIs.html) will give us some other KPIs we can use. If we want to use them we might need to create our own reports to automate the retrieval of this information. ITIL mentions among others these KPIs:

- Number of software defects in production, which is the number of bugs or software defects of applications (versions) that are in production
- Percentage of successful software upgrades (excludes full installations)
- Number of untested releases (i.e., not tested and signed-off)
- Number of urgent releases
- Average costs of release, where costs most likely are based on man-hours spent

Note The Information Technology Infrastructure Library (ITIL), is a set of practices for IT service management (ITSM) that focuses on aligning IT services with the needs of business. ITIL describes procedures, tasks, and checklists that are not organization-specific, used by an organization for establishing a minimum level of competency. It allows the organization to establish a baseline from which it can plan, implement, and measure. It is used to demonstrate compliance and to measure improvement.

Standard Reports

There are three reports in TFS 2012 that we can use to get information about our builds:

- Build quality indicators
- Build success over time
- Build summary report
The Build quality indicators (see Figure 29-1) report shows a summary of some important values for our builds. Using this data we can see whether we are close to releasing the build. Some of the information this report shows:

**Build Success Over Time**

Helps you track changes in the quality of the code that the team has checked in. Shows test results for the last build of each day.

![Build Success Over Time chart](image)

**Figure 29-1. Quality Indicators report**

- **Active bugs.** How many active bugs that existed at the time of the build.
- **Code churn.** The number of lines of code that has been added, removed, and changed in the check-ins before the build.
- **Code coverage.** How many percent of the code has been covered by tests.
- **Inconclusive tests.** The number of tests that did not succeed or were paused. If the build did not succeed, the tests are either not counted or counted as inconclusive.
- **Failed tests.** How many tests failed during the build.
- **Passed tests.** How many tests that were passed during the build.

The Build Success Over Time report (see Figure 29-2) shows us the status of the last build for each build category (a combination of build definition, platform, and configuration) run for each day. We can use this report to help us keep track of the quality of the code that we check in. Furthermore we can, for any day on which a build ran, view the Build Summary for that specific day.