

Configure the Velocity widget

04/04/2019 • 6 minutos para ler • Colaboradores

Neste artigo

[Prerequisites](#)

[Add the widget to your dashboard](#)

[Configure the Velocity widget](#)

[Required and recommended activities](#)

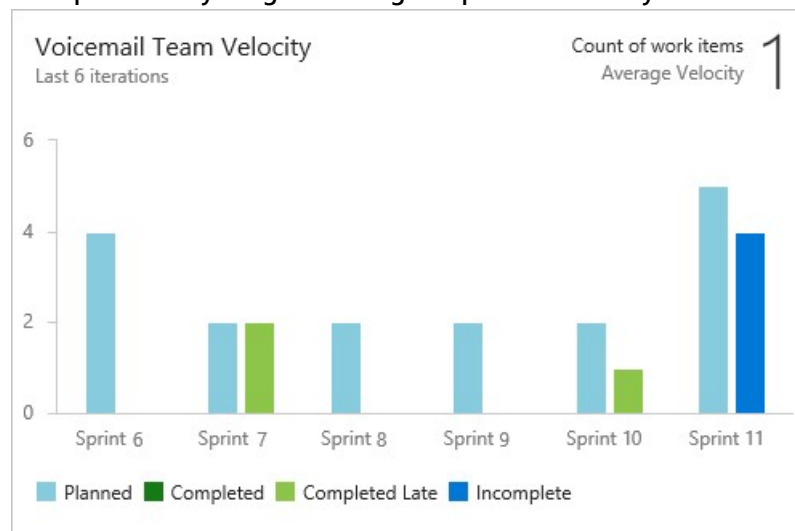
[Add other teams](#)

[Try this next](#)

Azure DevOps Services | Azure DevOps Server 2019

Teams track their velocity to help them determine how much work they can perform sprint-over-sprint. Velocity provides an indication of how much work a team can complete during a sprint based either on a count of work items completed or the sum of estimates made to Effort (PBIs), Story Points (user stories), or Size (requirements).

Example: Velocity widget showing six sprints of velocity



ⓘ Observação

The Velocity widget is based on the Analytics Service. The Analytics Service is generally available for Azure DevOps Services and in preview as an extension for Azure DevOps Server 2019. For TFS 2018 and earlier versions, you have access to the [velocity chart provided by the work tracking datastore](#).

Use this article to learn:

- ✓ How to configure the Velocity widget
- ✓ Required and recommended team activities to support velocity tracking

Once your team has completed a few sprints, they can use their velocity to [forecast](#) how much of the backlog they can finish within upcoming sprints. For usage guidance, see [Velocity metrics and usage guidance](#).

There are two velocity charts, the one you access by adding the Velocity widget to a dashboard, and the one [viewed from the backlog of a team](#). The Velocity widget enables you to view more sprints and additional

information than that provided by the velocity chart.

Prerequisites

- You must be a member of a project. If you don't have a team project yet, [create one](#).
- If you haven't been added as a project member, [get added now](#).
- To add a widget to a team dashboard, you need to be a member of the team. You must have **Basic** access or greater, have [dashboard permissions](#), or be a [team admin](#) or project admin. Default settings provide all team members with permissions.
- **Boards** must be enabled. If disabled, none of the work tracking Analytics widgets will display. To re-enable it, see [Turn an Azure DevOps service on or off](#).

Add the widget to your dashboard

1. If you haven't yet [added the Velocity widget to your dashboard](#), do that now.

Configure the Velocity widget

You configure your velocity widget for a single team. If you want to view the velocity for several teams, then you must configure a portfolio management team which rolls up from several teams. To learn more about teams, see [Add teams](#).

1. Choose the **...** actions icon and choose the Configure option to open the configuration dialog.

Modify the title, select the team, and then choose either the backlog level or work item type to track. Select whether you want to track a count of work items or a sum of a numeric field. The most common summed field is that of Effort, Story Points, or Size.

Configuration

Title

Voicemail Team Velocity

Size

3 x 2

Team

Voicemail

Work items

☒ Backlog

Features

☐ Work item type

Bug

Velocity

Count of work items

Number of Iterations

6

Advanced features

☒ Display planned work for iterations

Days past start date of iteration when planned work is final: 0

☒ Highlight work completed late

Days past end date of iteration after which work is late: 0

Save

Cancel

2. Specify the number of sprints you want to view. The default is 6 and the maximum is 15.
3. (Optional) Select the check boxes to show additional information for work completed later than planned for each sprint.

Displayed planned work for iterations: Check this box to display the amount of work planned for an iteration at the start of the iteration. This is useful for comparing your planned work to actual deliverables. By default, the count of planned work begins as of the start date of the iteration.

- **Days past start date of iteration when planned work is final:** Specify a number of days past the start date to count planned work. For example, if the first 2 days of an iteration are for planning, then you can enter "3", and planned work will be counted on the 3rd day.

For example, if the Iteration starts on 01/01/2018, and 3 backlog items are assigned to the iteration on 01/01/2018 end-of-day, then those 3 backlog item items will be considered as Planned. If your team doesn't complete planning until a few days into the iteration, then you can update the Days past start date of iteration when planned work is final.

ⓘ **Observação**

Work is considered Planned if it is assigned to the iteration as-of the Iteration Start Date.

Highlight work completed late: Work items marked complete after the iteration end date are considered to be completed late and will show as light green. This is useful for spotting a trend where work items are marked complete after the iteration is complete.

- **Days past end date of iteration after which work is late:** Specify a number of days past which a work item is considered late if it's status is still new or in progress.

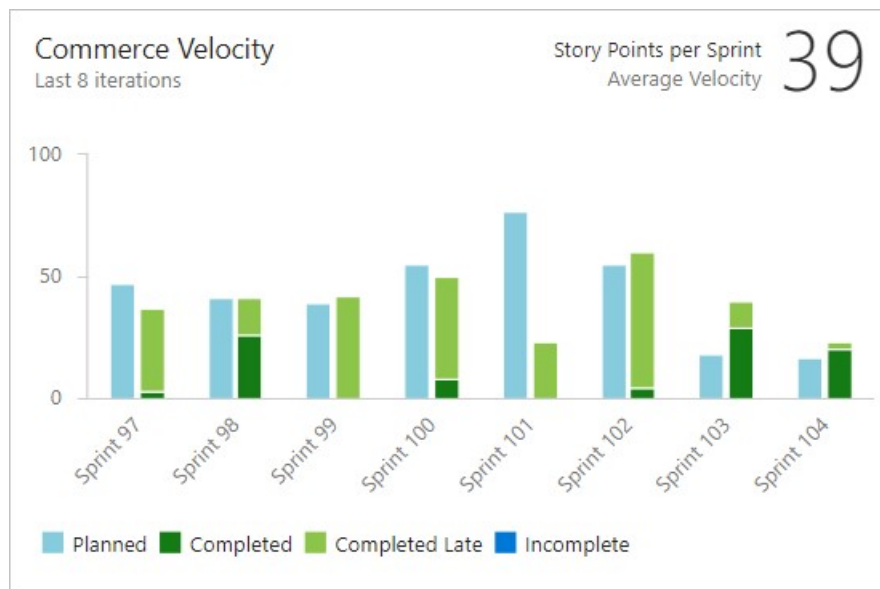
For example, entering 3 days will give the team 3 days after the end of an iteration to mark work items complete or done, before they are considered late.

ⓘ **Observação**

A work item is considered late when the work item's Completed Date is later than End Date of the Iteration the work item is *currently* assigned to.

It will take into account the value you enter for *Days past end date of iteration after which work is late*.

4. Choose **Save** when done. The following image shows Velocity based on Story Points and 8 sprints of data.



Required and recommended activities

For your team to gain the greatest utility from the velocity chart, follow these required and recommended tasks.

Required:

- [Define iteration paths \(aka sprints\) and configure team iterations](#) Sprints should be of the same duration.
- [Define and estimate backlog items](#). If you work from your team's backlog, the items you create will automatically be assigned to the current sprint (Iteration) and to your team's default Area Path.
- Update the status of backlog items once work starts and when completed. Only backlog items whose State maps to a metastate of In Progress or Done will show up on the velocity chart or velocity widget.

Recommended:

- Define and size backlog items to [minimize variability](#).
- Determine how your team wants to [treat bugs](#). If your team chooses to treat bugs like requirements, bugs will show up on the backlog and be counted within the Velocity chart and forecasting.
- [Set your team's area path](#). The forecast tool will forecast those items based on your team's default settings. These settings can specify to include items in area paths under the team's default or exclude them.
- Don't create a hierarchy of backlog items and bugs. The Kanban board, sprint backlog, and task board only show the last node in a hierarchy, called the leaf node. For example, if you link items within a hierarchy that is four levels deep, only the items at the fourth level appear on the Kanban board, sprint backlog, and task board.
Instead of nesting requirements, bugs, and tasks, we recommend that you maintain a flat list-only creating parent-child links one level deep between items. Use [Features to group requirements or user stories](#). You can quickly map stories to features, which creates parent-child links in the background.
- At the end of the sprint, update the status of those backlog items that the team has fully completed. Incomplete items should be moved back to the product backlog and considered in a future sprint planning meeting.

Add other teams

If you work with several teams, and each team wants to work with their own backlog view, velocity chart, and

forecast tool, you can [add teams](#). Each team then gets access to their own set of Agile tools. Each Agile tool filters work items to only include those whose assigned area paths and iteration paths meet those [set for the team](#).

Try this next

Velocity guidance