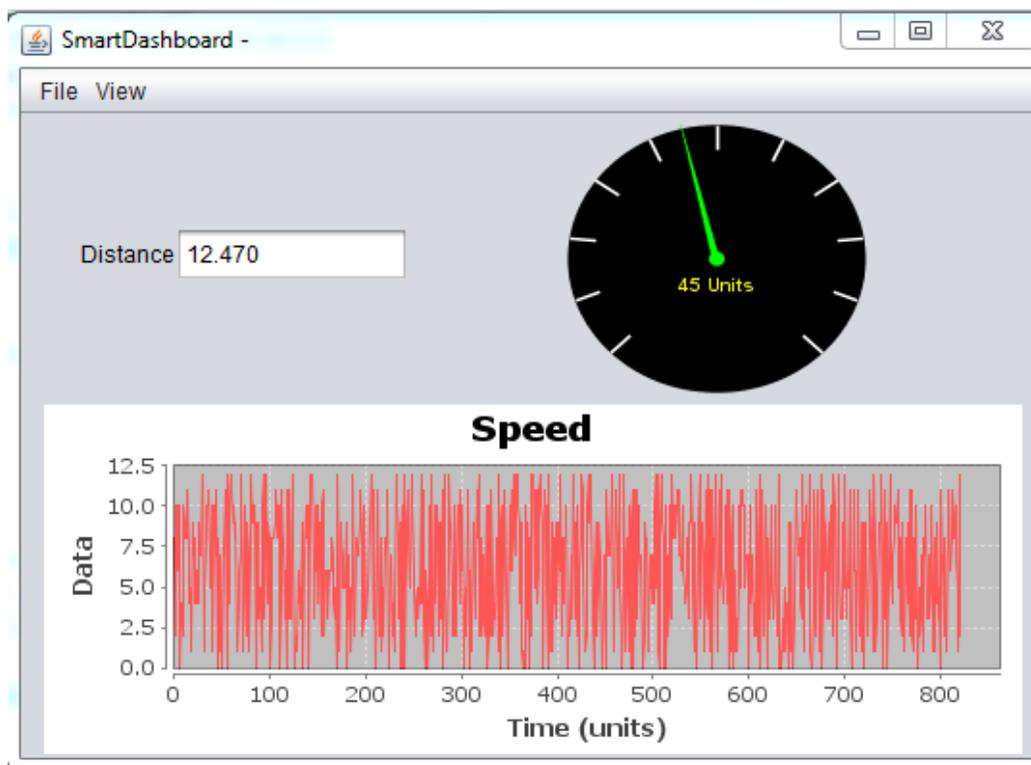# Getting Started with the SmartDashboard

The SmartDashboard typically runs on the Driver Station computer and will do two functions:

1. View robot data that is displayed as program status as your program is running.
2. View sensor data and operate actuators in Test mode for robot subsystems to verify correct operation.

The switch between program status and test modes are done on the Driver Station.
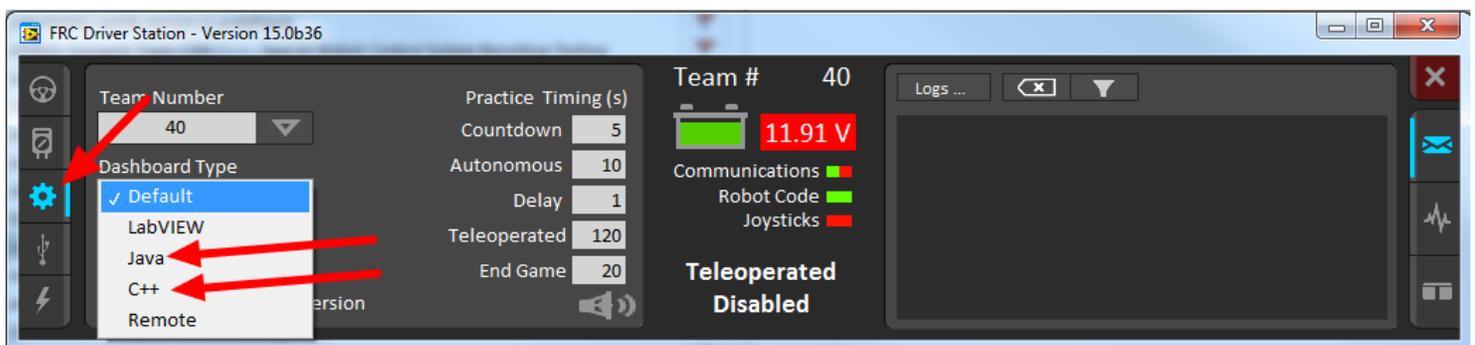
What is the SmartDashboard?

The SmartDashboard is a Java program that will display robot data in real time. The SmartDashboard helps you with these things:

Getting Started with the SmartDashboard

- Displays robot data of your choice while the program is running. It can be displayed as simple text fields or more elaborately in many other display types like graphs, dials, etc.
- Displays the state of the robot program such as the currently executing commands and the status of any subsystems
- Displays buttons that you can press to cause variables to be set on your robot
- Allows you to choose startup options on the dashboard that can be read by the robot program

The displayed data is automatically formatted in real-time as the data is sent from the robot, but you can change the format or the display widget types and then save the new screen layouts to be used again later. And with all these options, it is still extremely simple to use. To display some data on the dashboard, simply call one of the SmartDashboard methods with the data and its name and the value will automatically appear on the dashboard screen.

Installing the SmartDashboard

The SmartDashboard is now packaged with the C++ and Java language updates and can be launched directly from the Driver Station by selecting the **Java** or **C++** buttons on the **Setup** tab.

Note: The 2015 Kickoff release of the Driver Station contains a bug which prevents this setting from sticking across restarts of the DS Software. See this article for details on how to get the DS to launch the SmartDashboard when it starts.

Note: If using the Classmate PC or other PC where the DS is run from a separate account from the C++\Java tools install (e.g. Driver and Developer) the buttons shown above may not work. You can utilize the SmartDashboard in this type of setup in one of two ways. The first way is to set the type to **Default** and modify the DS INI file to launch the appropriate dashboard, details can be found in this article. The second option is to copy the C:\Users\Developer\wpilib\tools directory to C:\Users\Driver\wpilib\tools. When using this second method it is recommended to make sure that the Dashboard under both the Driver and Developer accounts point to the same save file (see Locating the Save File) below.

Getting Started with the SmartDashboard

Configuring the Team Number
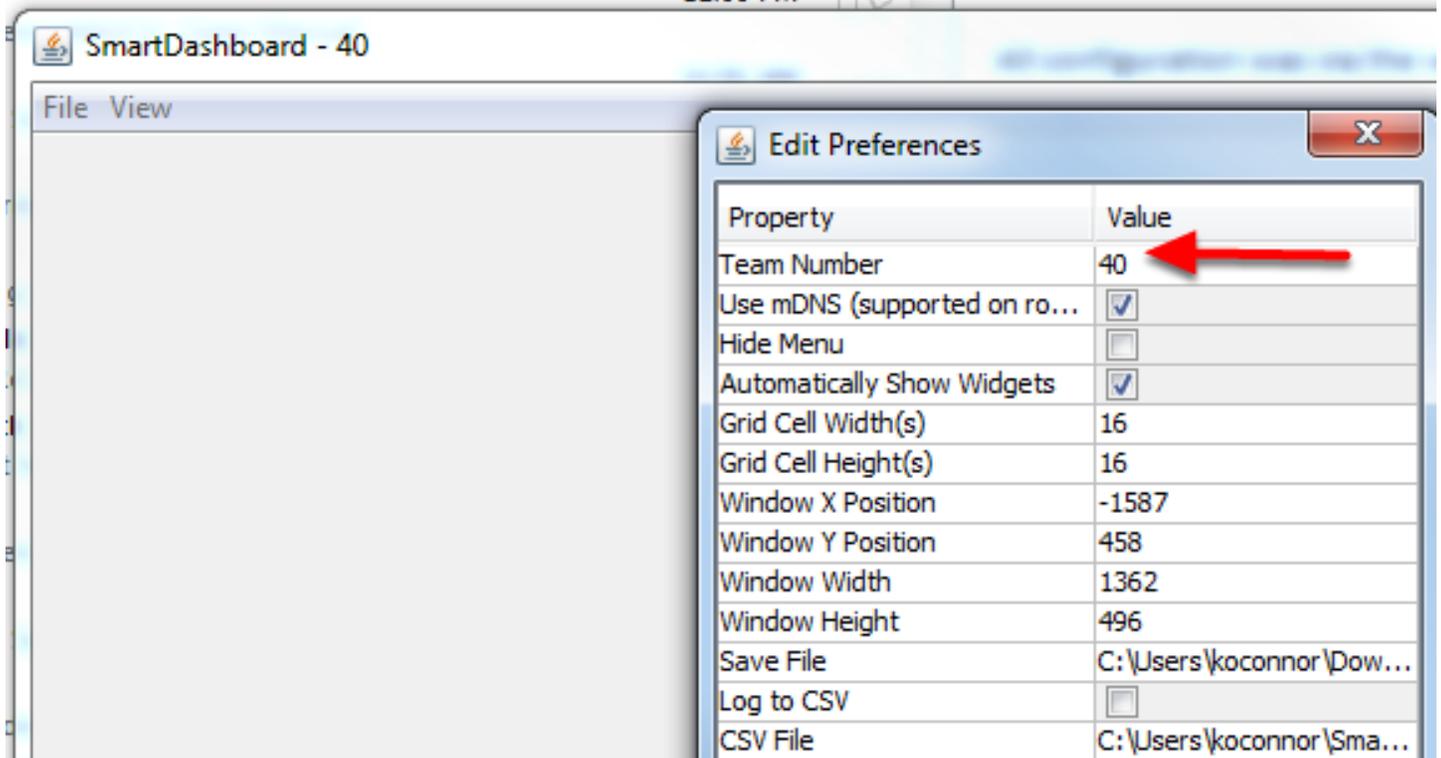
The first time you launch the SmartDashboard you should be prompted for your team number. To change the team number after this: click **File > Preferences** to open the Preferences dialog. Double-click the box to the right of **Team Number** and enter your FRC Team Number, then click outside the box to save. **Note that the SmartDashboard will take a moment to configure itself for the team number, do not be alarmed.**
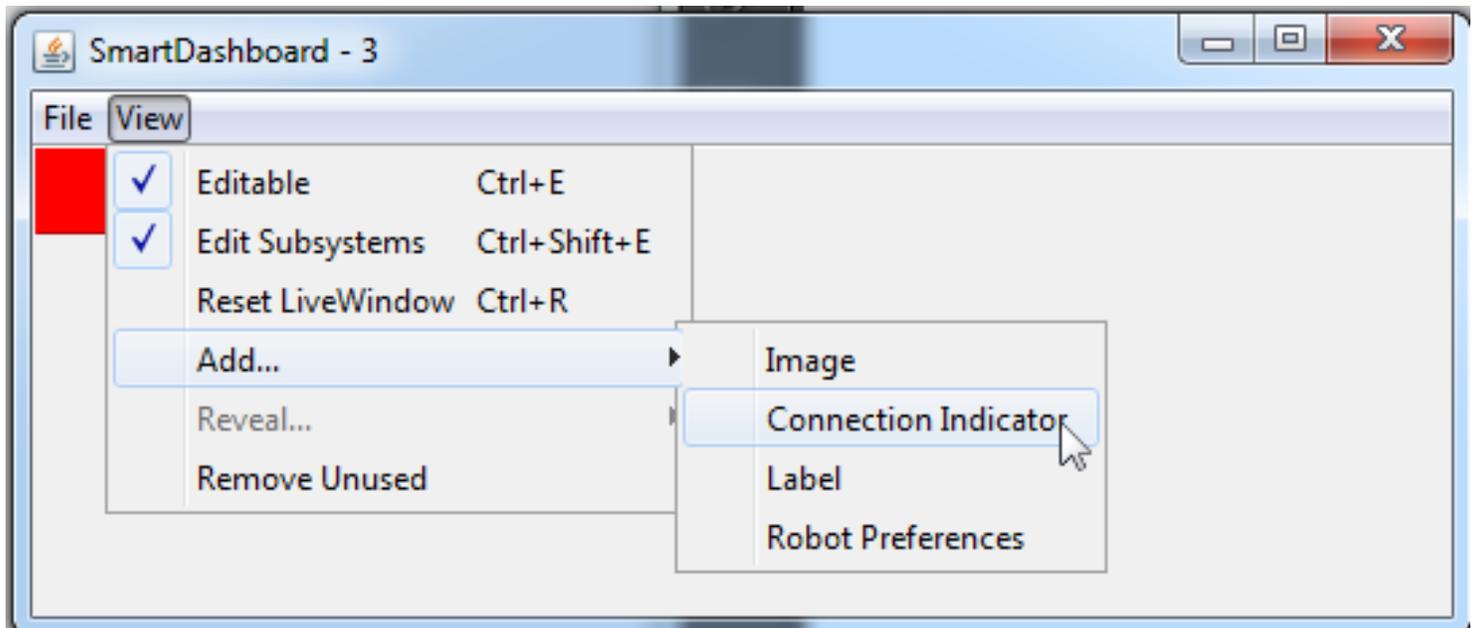
Getting Started with the SmartDashboard

Locating the Save File
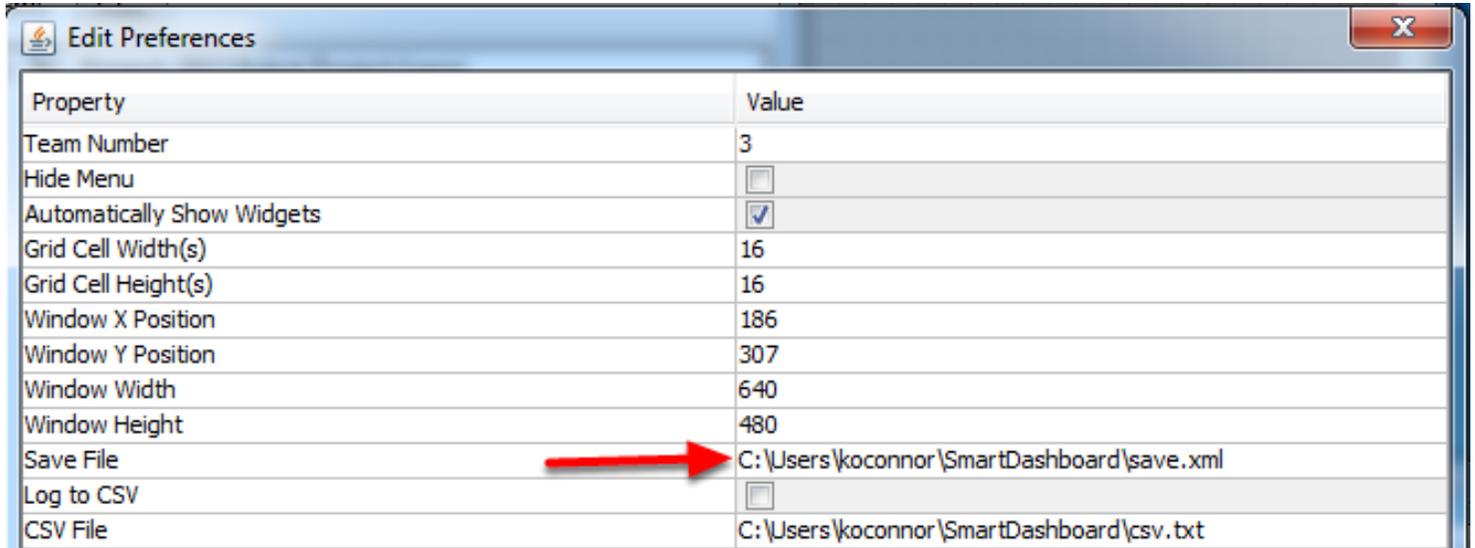
Users may wish to customize the save location of the SmartDashboard. To do this click the box next to **Save File** then browse to the folder where you would like to save the configuration. **It is recommended that this folder be inside the users home directory and not inside the sunspotfrcsdk or workbench directories.** Files saved in the installation directories for the WPILib components will likely be overwritten on updates to the tools.

Adding a Connection Indicator

Getting Started with the SmartDashboard

It is often helpful to see if the SmartDashboard is connected to the robot. To add a connection indicator, select **View > Add > Connection Indicator**. This indicator will be red when disconnected and green when connected. To move or resize this indicator, select **View > Editable** to toggle the SmartDashboard into editable mode, then drag the center of the indicator to move it or the edges to resize. Select the **Editable** item again to lock it in place.

Adding Widgets to the SmartDashboard
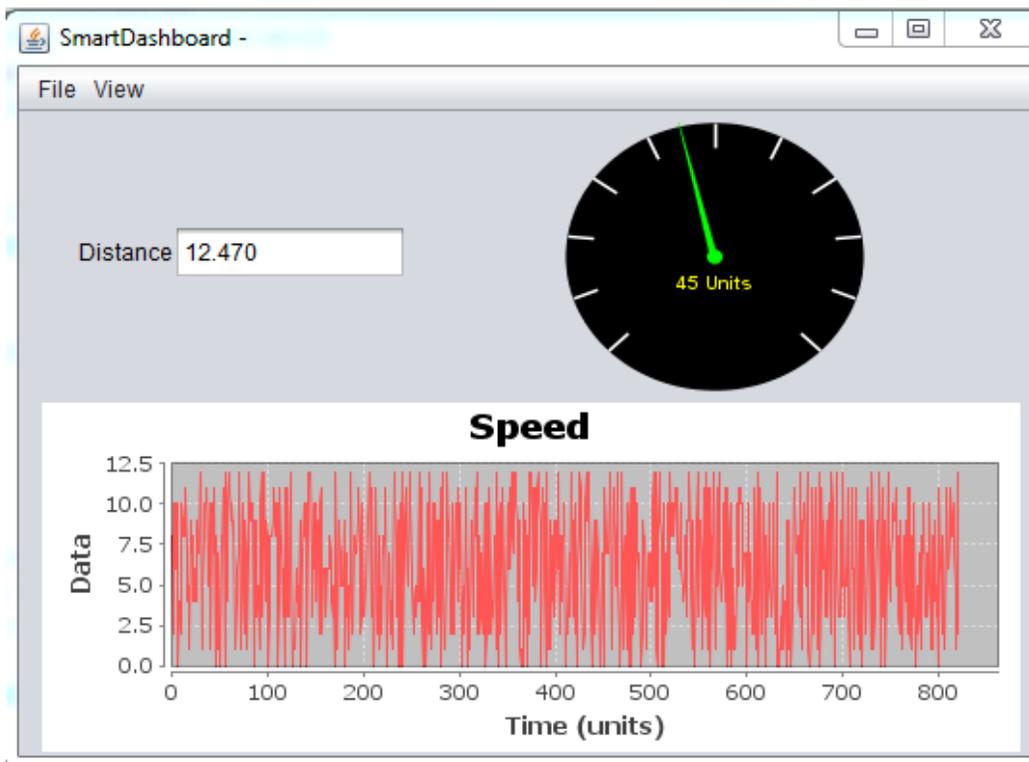
Widgets are automatically added to the SmartDashboard for each "key" sent by the robot code. For instructions on adding robot code to write to the SmartDashboard see [Displaying Expressions from Within the Robot Program](#).