

# Xilinx Opens a Tcl Store

by **Greg Daughtry**  
Director of Product Marketing  
Xilinx, Inc.  
[greg.daughtry@xilinx.com](mailto:greg.daughtry@xilinx.com)



# Open-source repository for sharing Tool Command Language scripts is up and running at GitHub.com.



Over the last five years Xilinx has had a strategic focus on design methodology and tools to address productivity, to accelerate the design cycle and to help bring products to market faster by providing the industry's most advanced and comprehensive development environment.

Even with the productivity improvements of the next-generation Vivado® Design Suite combined with the comprehensive UltraFast™ Design Methodology, designing with today's All Programmable devices can be challenging. Designers must integrate hundreds of highly parameterized IP cores, hundreds of thousands of placeable objects and multiple millions of logic cells with Xilinx® All Programmable FPGAs, 3D ICs and SoCs. There are an infinite number of permutations to grapple with as designers push the boundaries with complex designs.

With the release of Vivado 2014.1 in April, Xilinx is taking another large step forward in designer productivity by hosting an open-source repository for sharing Tool Command Language (Tcl) code. This repository, called the Xilinx Tcl Store, will make it a lot easier to find and share Tcl (pronounced "tickle") scripts that other engineers have developed. With the power of Tcl, these scripts can extend the considerable core functionality of the Vivado Design Suite, enhancing productivity and ease of use. The Tcl Store is open to the user community to contribute to the greater good of all designers by publishing Tcl code that others might find useful.

## DESIGNS GROWING MORE COMPLEX

The Vivado Design Suite was built on an open, scalable data model. As an open system, one of the keys to enabling productivity is making the tools smarter, and providing more customization choices and analysis capabilities so the designer can be better in-

formed and drive the tools to provide optimal implementations.

Since the release of the Vivado Design Suite in 2012, there has been an explosion of Tcl scripting to perform tasks both small and large. It's increasingly important for designers to understand and utilize Tcl, since this is the basis for Vivado's XDC constraint language.

The Tcl commands allow you to develop and scrub timing constraints interactively, which saves compilation time and debug effort. The core commands allow object queries that can be used for custom reporting, and that can execute very elaborate tool control. The Vivado Design Suite makes it possible to also develop your own DRC and linting checks, along with highly customized flows to achieve better quality of results or faster run-times. Tcl also enables designers to make targeted design changes through engineering change order (ECO) operations.

The increased productivity provided with Tcl, ease of creation and readability make this language prime for the sharing of useful code. Up to now this sharing has largely occurred on an ad hoc basis, via e-mail and user forums. Some companies have established their own internal libraries of Tcl for use within their projects.

Now Xilinx is taking Tcl sharing to the next level with its new Xilinx Tcl Store.

## WELCOME TO THE TCL STORE

The Xilinx Tcl Store provides examples of how to write custom reports, control specific tool behavior, make custom netlist changes and integrate with third-party electronic design automation (EDA) tools such as simulation, synthesis, timing and power analysis, and linting tools.

Natively accessed from the Vivado Integrated Design Environment (IDE), the Tcl Store enables users to select and install collections of Tcl scripts called "apps" directly from within the tool. Once installed, these apps have commands that appear just like built-

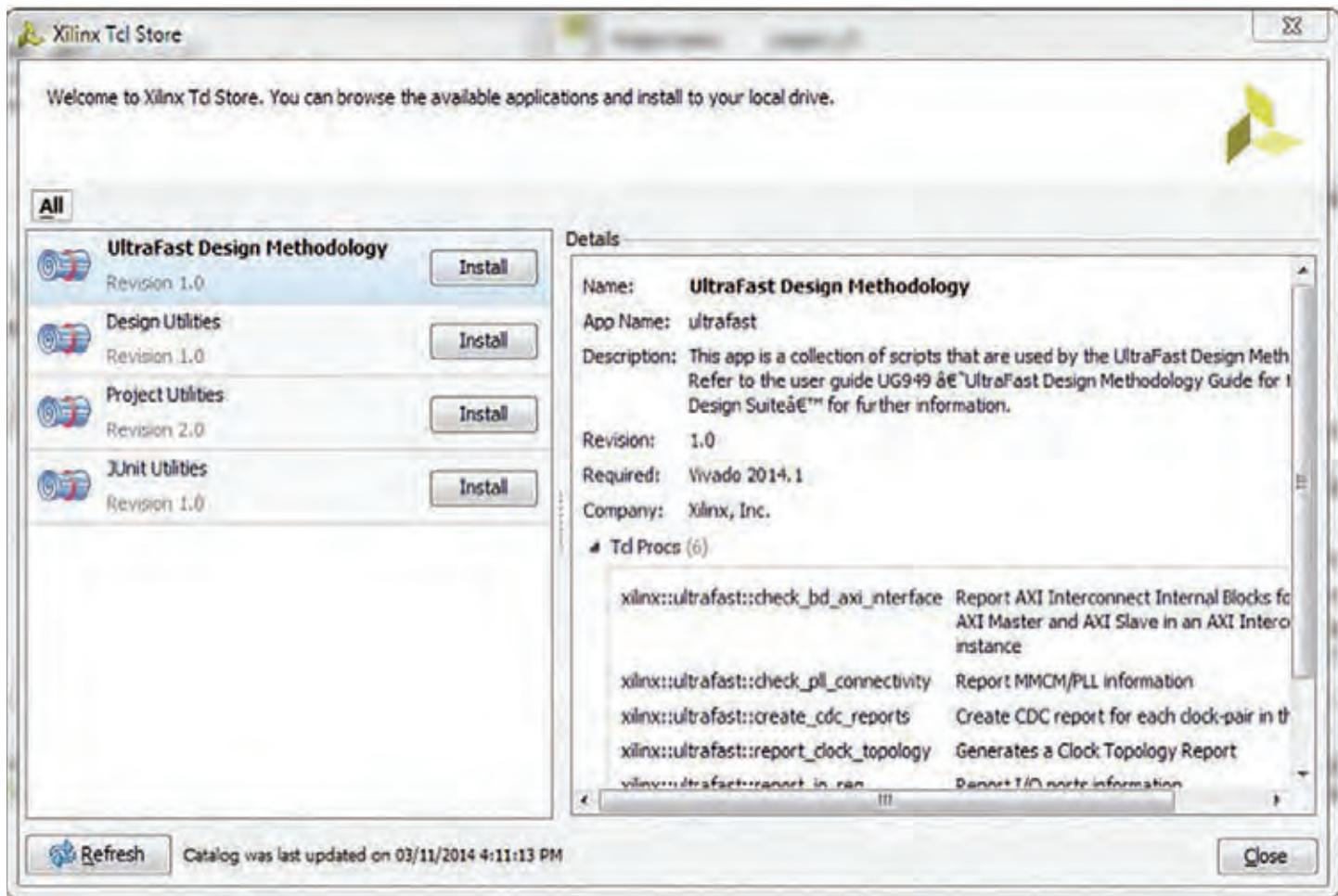


Figure 1 – The Tcl Store dialog box in the Vivado IDE allows installation of apps and browsing of the commands.

in Vivado Design Suite commands, right down to the help infrastructure. Vivado Design Suite supports different versions of apps using standard package facilities of Tcl, so if a newer version is released you can choose to upgrade with a single mouse click.

The Xilinx Tcl Store is intended to make it easier to find and use well-crafted Tcl scripts developed and supported by the user community, in the same manner as Linux development. Tcl scripting is a little more advanced than selecting IDE buttons. However, it is easy to learn. Documentation and user guides provide details on specific commands from the Tcl API and can be found on [xilinx.com/support](http://xilinx.com/support).

Let's take a closer look at the infrastructure for installing and using Tcl apps from the Xilinx Tcl Store.

## INSTALLATION AND USE

Designers can access the Xilinx Tcl Store by means of an icon on the Getting Started page when you first launch the Vivado IDE. Alternatively, you may also go to the Tools Menu and select the "Xilinx Tcl Store" menu option. This will bring up the repository dialog box, which will give you a list of apps available to install (Figure 1).

As you browse the list of apps, within each app there is a list of commands (called "procs" in Tcl) that are available for execution. You will see a description of each app, and of each proc within the app, to get an idea of what it does. Click on the install button to install the app and register it so that it now shows up like native Vivado Design Suite commands. Once an app is installed, each time you start the Vivado

Design Suite it loads automatically—there is no need to install the app each time you start a new session.

Procs have a naming convention that uses a facility in Tcl called namespaces. The names of the commands may seem a little more complex than normal Tcl commands, and have ":" characters embedded in them. For example, xilinx::ultrafast::check\_pll\_connectivity runs some connectivity checks on the clock-modifying blocks in Xilinx devices. The naming conventions serve to make sure the Tcl code is unique and that a proc in one app does not conflict with another proc by the same name in another app. Namespaces are a standard feature of Tcl.

To execute an app command, type in the fully qualified name of the proc in-

# Usage of Tcl apps in the Xilinx Tcl Store is meant to be easy and simple. Xilinx's goal is to encourage use and sharing among development teams around the world to improve productivity.

cluding the namespace, and optionally pass in any required arguments, just like other Tcl commands. Since the commands are using standard namespaces, you can also choose to import the commands into the global namespace. The strategy will work fine if there are no conflicts between any other command names. This will allow you to omit the namespace qualifier and use the proc name alone. In the example above, if you imported the UltraFast app into the global namespace, you could call the check\_pll\_connectivity command directly without the namespace qualifiers.

Designers can uninstall apps with a single click of the “Uninstall App” hyperlink within the details section of the app. There is also a “Refresh” button to update the catalog. The Tcl Store catalog is hosted on a third-party website that provides the ability to push out updates to app revisions independently of Vivado Design Suite releases. If the catalog is refreshed, the Vivado tools will perform a lightweight synchronization of the list of apps. If an updated version of an installed app is available, use the “Update” button to acquire it. The Vivado Design Suite will copy and sync the latest version of the app and install it. To avoid configuration control issues, upgrades are only installed at the designer’s request. For those who are concerned about security and would prefer to keep the Vivado Design Suite from syncing outside of their network firewall, there is a parameter to disable the catalog synchronization.

Usage of Tcl apps in the Xilinx Tcl Store is meant to be easy and simple. Xilinx’s

goal is to encourage use and sharing among development teams around the world to improve productivity. Only the latest version of any given app is displayed and designers can only install or upgrade to the latest supported version. Of course, the best way to have good usage is to ensure that there is a rich library of useful code. Xilinx has seeded the repository with a collection of helpful utility and integration scripts that you can peruse as good examples of how to build your own reusable Tcl scripts.

## CONTRIBUTING TO THE TCL STORE

There are two ways to contribute to the Tcl Store and make your script available to all Vivado Design Suite users. The first is to modify an app that already exists. The second is to develop and submit a request for a new app. To contribute code to the repository, you need to have some level of comfort with software development tools for revision control, or at least a willingness to learn.

Each app is controlled by a single person, usually the person who authored most of the code, referred to as the “app owner.” The Xilinx repository as a whole is controlled by Xilinx, and the company maintains a process for releasing the apps into the public domain to enforce basic consistency across the apps. Xilinx employees will perform a “gatekeeper” role to ensure quality.

The “contributor” who wishes to modify an existing app or add a new one will work with the gatekeeper and app owner for the submission, consistent with the process on other open-source

projects. A wiki on the site where the code is hosted documents this process.

Basic requirements will be enforced for all code submissions. Xilinx has attempted to keep this list—which is subject to change—as small as possible, while still ensuring a reasonable user experience. Here is the list of basic app requirements you need to adhere to:

- Follow basic coding-style guidelines by using procs with arguments that do not use or access global variables.
- Include basic documentation inside the proc that describes what it does, what the arguments are and what it returns.
- Make sure code passes a basic syntax check, and also passes a linting tool that is provided as a part of the Vivado Design Suite.
- Provide a minimum of one basic test for each proc that ensures the code at least runs and does what is expected.

## THE TCL STORE ON GITHUB

The Xilinx Tcl Store is hosted on a third-party website called GitHub.com. The store uses revision-control tools to ensure distributed development happens in a controlled way. The key to this process is Git, a popular open-source, distributed revision-control tool that is commonly used for Linux. To access the repository for contribution and testing, you register for a free account on GitHub.com, and install and set up Git. GitHub provides an installation of Git tools for Windows

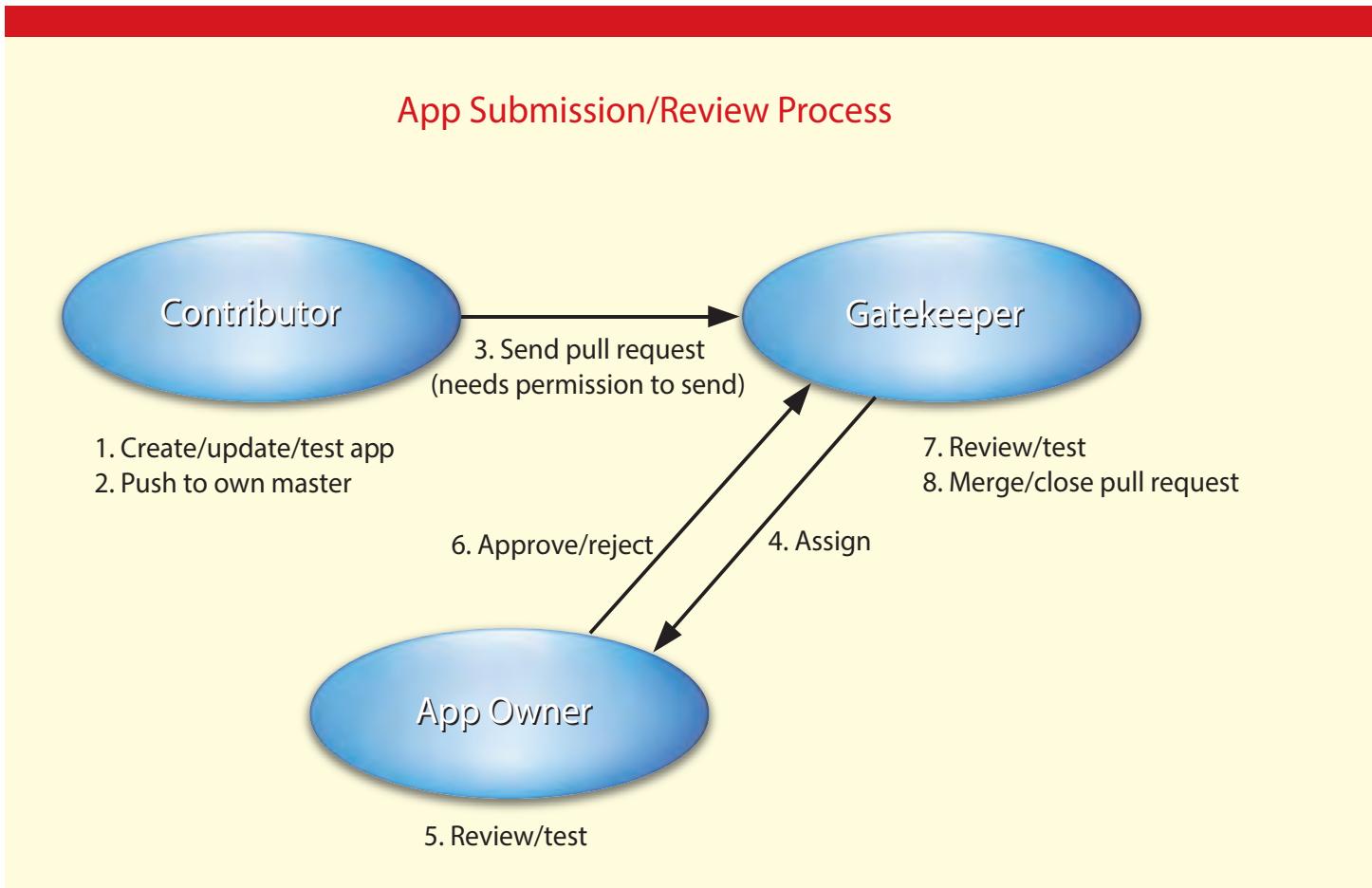


Figure 2 – Workflow of the Xilinx Tcl Store submission process passes through several discrete steps.

PCs. Linux machines typically already have it or can install it through standard packages. GitHub provides tutorials to help you get started with Git.

Once you have a GitHub account, here are the steps for contributing to the Tcl Store repository:

1. Clone the Xilinx Tcl Store master repository. This creates a local copy that is your sandbox and it allows you to develop locally and test without impacting others.
2. Place your new code in the correct directory, following the established guidelines based on app name and company or GitHub username. Use standard Git add commands.
3. Use Vivado Design Suite in the local repository, and call a few commands that are necessary for registering the

code and generate a catalog.xml file. This is one of three files that you will need. The others are a package index and a Tcl index.

4. Open Vivado Design Suite in another location, point to the local repository and test your apps. Run the linter and your local tests until you are comfortable that all is working correctly.
5. Commit your changes and provide a message briefly documenting them.
6. Send an e-mail requesting permission to contribute to [tclstore@xilinx.com](mailto:tclstore@xilinx.com). Indicate whether you'd like to create a new app and what you'd like to call it. If you'd like to modify or contribute to an existing app, please indicate that; you will need permission from the app owner.

7. Go to GitHub.com using a Web browser and issue a pull request. This formally initiates the process of merging your contributions into the repository. Work with the gatekeeper and app owner as appropriate to resolve any issues through GitHub and e-mail.
8. Congratulations! It feels good to help your fellow designers.

Figure 2 shows a basic diagram of the workflow showing the submission process.

### THE FINE PRINT

The Xilinx Tcl Store is open source, and there is no facility to monetize or charge for contributions. Apps contributed to the Tcl Store are made freely available for derivative works through a BSD license commonly used in open-source projects.

Contributions to the repository will include a version of the BSD license with each app in order for it to be accepted and published publicly. If a company or user does not wish to release their intellectual property into the public domain, Vivado Design Suite does support local versions of the repository via the same mechanism that is used for testing prior to issuing a pull request.

Furthermore, since the project uses GitHub for hosting, submitters must agree to the GitHub terms of service when you register for an account, as this is a third-party service.

The apps in the app store are developed and supported by the user community. This means that Xilinx technical support has not received training on this functionality and will not be able to answer questions about Tcl code. Please direct support questions for these apps to the Xilinx user forums. If a bug or issue exists in a piece of code, you can file and track it directly in GitHub.com projects. Since this is an open-source development model, users are encouraged to fix these issues and improve the experience for the overall good of other users—just like Linux.

## ROAD MAP

The Vivado 2014.1 introduction of the Tcl Store is just the beginning. Xilinx will be improving the Tcl Store this year by implementing the ability to search the descriptions of the apps and procs to make it easier to find functions. We will be providing a way to browse and view the source code without having to install the app. In addition, we intend to provide a review mechanism where users can specify a rating of one to five stars, and optionally provide a written review. This will give people feedback mechanisms on the more popular submissions.

We will also make use of better categorization with filtering capability based on the categories of apps, for example simulation, synthesis, implementation, project and netlist utilities. As the repository grows, we may institute more groupings and extend the taxonomy of apps to reflect the contributions.

We want to make it as easy as possible to contribute apps, so we may look for ways to allow people to submit Tcl scripts by e-mail with a minimal support burden and no need to go through GitHub. The uncontrolled nature of such a process would not be coupled with the current installation scheme, and would perhaps be best suited for examples.

Thousands of designers around the world are using the Vivado Design Suite and hundreds of companies have adopted the UltraFast Design Methodology. The Xilinx Tcl Store will continue to increase designer productivity by providing a new open-source project between Xilinx, its partners and our customers aimed at sharing Tcl scripts. ☀

## INFORMATION AND RESOURCES

To request GitHub account access and to see tutorials on Git and GitHub, go here:

<https://github.com/>

The Xilinx Tcl Store code repository as well as a wiki that documents how to contribute are located here:

<https://github.com/Xilinx/Xilinx-TclStore>

The Xilinx Tcl Store wiki contains detailed information concerning the contribution process:

<https://github.com/Xilinx/Xilinx-TclStore/wiki/Xilinx-Tcl-Store-Home>

UG 894, the “Using Tcl Scripting Guide,” contains information on the general scripting capabilities of the Vivado Design Suite:

[http://www.xilinx.com/support/documentation/sw\\_manuals/xilinx2013\\_4/ug894-vivado-tcl-scripting.pdf](http://www.xilinx.com/support/documentation/sw_manuals/xilinx2013_4/ug894-vivado-tcl-scripting.pdf)

UG 835, “Tcl Command Reference,” contains information on all of the native Tcl commands available in Vivado Design Suite:

[http://www.xilinx.com/support/documentation/sw\\_manuals/xilinx2013\\_4/ug835-vivado-tcl-commands.pdf](http://www.xilinx.com/support/documentation/sw_manuals/xilinx2013_4/ug835-vivado-tcl-commands.pdf)

**KNOWLEDGE RESOURCES**  
Solutions GmbH

**FPGA Module**

Zync 7Z020 based  
192 PLI/O on connector  
MIO\_1 on connector  
Dual QSPI config on board  
FOM compliant

**Evaluation Board**

Accepts FOM modules  
4 expansion connectors  
- Lan  
- USB  
- UART to USB  
- Micro SD  
and much more

**Expansion Boards**

Motor driver      DVI out  
3 band ISM RF      5MP camera  
- More in the pipeline

**Support**

Library components  
Reference designs  
Linux BSP  
Heat-spreader solution  
Engineering support on demand

[www.knowres.com](http://www.knowres.com)