

Creating and Using Packages

What is a package?

A package is a collection of checks. A check can be included in many packages. While the system comes pre-configured with a number of packages, their primary strength is that the user can easily create and alter packages without any fear of corrupting the functionality of the system as a whole.

How do I create a package?

To create a package, you must first find the **bps_setup.tcl** file in your installation directory, and copy it to a local directory called **Blue Pearl Software, Inc\Blue Pearl Visual Verification Environment**. In Windows 7, this new directory should be located in your standard **Documents** library. In Linux, this new directory should be located within a directory called **~/Documents**. And yes, Linux users, that means you must create a directory with spaces in its name. Note that this is only the most local possible location; see **Section 2.1** of the **Analyze Reference Manual** for the complete search path.

Open your copy of the **bps_setup.tcl** file with your preferred text editor and find any occurrence of the string `add_package`. This will be in the context of a line that looks like this:

```
BPS::add_package {Coding Style}
```

This is followed by a list of similar commands of the form:

```
BPS::add_check_to_package -check {HCCC} {Coding style}
```

In this example, HCCC is the rule mnemonic. The list is similar to what you see when you open the **Analysis Settings** dialog, click the **Packages** radio button, and expand the **Coding Style** entry, as shown below:

- ▾ Coding style
 - HCCC - Do not hard-code constants
 - FANOUT - Maximum fan-out
 - NS_NAME - Next state names
 - ALN - Use underscore for active low signals

This illustrates a couple of points. The first is, the file includes the mnemonic label, but the GUI also includes a description. The file includes HCCC, and the corresponding line in the GUI is **HCCC – Do not hard-code constants**. Clearly, the additional labeling in the GUI came from somewhere else. If you were to try to include anything other than the mnemonic in the **bps_setup.tcl** file, the program would not run. Also, note that the list in the GUI is shorter than the list in the file. You will find the rest of the list when you open the **Design Settings** dialog and select the **Load Checks** menu. Some checks run during the *Load* phase, and some during the *Analysis* phase, and a few run in both.

Add to your copy of the **bps_setup.tcl** file something like the following:

```
BPS::add_package {Your new package}
  BPS::add_check_to_package -check {STP} {Your new package}
  BPS::add_check_to_package -check {STATE_TRANSITION} {Your new package}
  BPS::add_check_to_package -check {UNREACHABLE_STATE} {Your new package}
  BPS::add_check_to_package -check {TERM_STATE} {Your new package}
  BPS::add_check_to_package -check {UCCN} {Your new package}
  BPS::add_check_to_package -check {SLCC} {Your new package}
  BPS::add_check_to_package -check {BEC} {Your new package}
  BPS::add_check_to_package -check {TLIN} {Your new package}
```

In this example, the package name that will appear in the GUI is “Your new package,” just as the name of the default package used as an example is “Coding Style.” The checks that are part of the package are listed in commands immediately under the package name, one per line. The packages appear in the same order in which they appear in the file. If you want your custom package to appear at the top of the list, put it as the first package in the **bps_setup.tcl** file.

How do I use a package that I’ve created?

Once you’ve edited and saved your copy of the **bps_setup.tcl** file, exit from the **Blue Pearl Software Suite** and re-start it, then invoke your custom package just as you would any other. In this case, we’ve included both *Load* and *Analysis* checks in the new package, so our list is split between the **Design Settings -> Load Checks** and the **Analysis Settings -> Analysis Checks**, like this:

- ▾ Your new package
 - UCCN - Use uppercase constants/definitions
 - SLCC - Separate lines for commands
 - BEC - Big endian checks
 - TLIN - Avoid gate instances at the top level
- Load Checks
- ▾ Your new package
 - STP - FSM Analysis
 - STATE_TRANSITION - Report state transitions
 - UNREACHABLE_STATE - Report Unreachable states (if STP set)
 - TERM_STATE - Report Terminal states (if STP set)
- Analysis Checks

In most cases, you’ll probably want to run an analysis using only the checks in your package. Display the new package as shown, click the **Uncheck All** button, then check the box in front of the new package and run your analysis. In the examples shown above, the checkbox next to the new package is filled in, indicating that some but not all the checks in that package are selected. (These graphics are from Windows. In Linux, the checkbox may have a gray check instead of being filled in.)