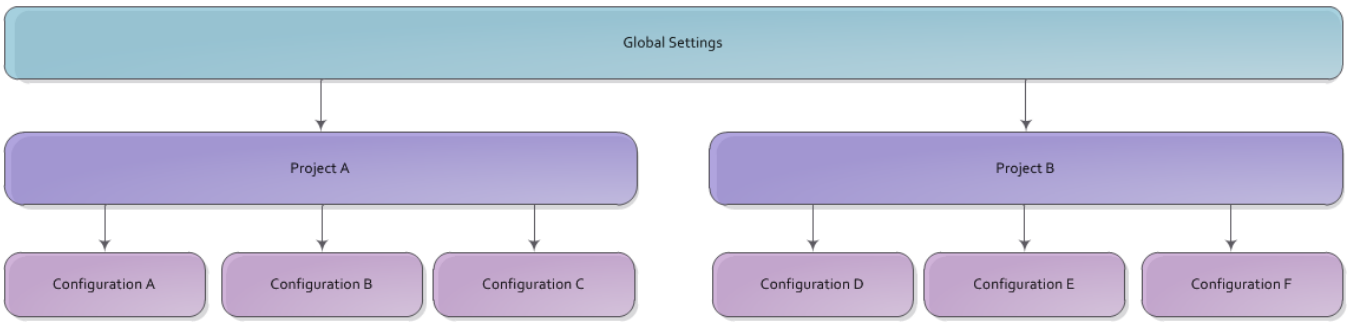
Global Settings, Projects & Configurations

Within the Continua CI environment there are three hierarchy levels: Global, Project and Configuration (see figure 1). This structure is similar to object oriented programming where children inherit settings from their parent, however they can also override these settings.

If you have properties set on all three levels of Continua CI, the lowest child will take precedence. For example, if you have security permissions set at the global, project and configuration level and a user views a particular configuration, Continua CI will always use the permissions created on the configuration.

If there is ever a conflict between multiple properties over multiple levels, Continua CI will always use the value on the lowest level. ie. Configurations take precedence over projects or global settings.

Figure 1: Demonstrates the relationship between Configurations, Projects and any settings that are set Globally. Each level of the tree can override the settings it inherits from its parent



Global

Anything created globally can be accessed by all projects and configurations in Continua CI. Many elements of Continua CI can be set globally including:

- Security permissions. By setting permissions at the global level you can create a blanket security policy for Continua CI.
- Variables. Creating variables globally allows all projects and configurations to reference the same variable with the same value.
- Continua CI Repositories. Creating a repository at the global level allows all configurations to access the same data in the same Version Control System (VCS). This means that if anything regarding your VCS changes, then you only need to make changes to one repository.

Project

Project acts as a container for configurations and enable Continua CI to group configurations together and apply settings across all configurations in that project. Anything created at the project level can be accessed by all configurations within that project. Projects have access to anything set on the global level but do not know about any settings in other projects.

Projects have a one-to-many relationship with configurations where each project can contain multiple configurations, however each configuration can only exist in one project.

For example, Project settings allow you to:

- Set user permissions that apply to all configurations in a certain project.
- Create repositories that are only accessible to configurations in a certain project.
- Override global variables for all configurations in a certain project.
- Create project specific cleanup policies.

Configuration

A configuration is the defining component of Continua CI and the continuous integration process. Anything set on the configuration will take precedence over anything set globally or on the configuration's project. Configurations are individual entities that do not know about other configurations. For example, if you add a repository onto configuration A, configuration B cannot find that repository or access it. If you need multiple configurations to share repositories, variables etc. then create them either at the project or global level.

Configurations contain all the information that is needed to complete builds, including:

- The Action Work Flow. This work flow is where you specify the individual actions that will run when you execute your build process. Using the Stage Editor you can define multiple build stages per configuration and set a variety of actions on these stages.
- **How a configuration is triggered**. Configurations can be fired either manually through the Continua CI user interface or by creating a Trigger on the configuration. Triggers can be set to fire in the following ways:

    - Fire the configuration when a checkin is made to a source control repository.
    - Fire the configuration at a certain date or time.
    - Fire the configuration after another Configuration has completed.
- **Continua CI Repositories**. Source control repositories, such as Subversion and Mercurial, can be linked to configurations so that a repository's activity can be monitored and in turn, trigger builds.
- **Build Conditions**. Conditions can be set on configurations to determine what will trigger a build. For example, configurations can be set to fire only if the previous configuration completed its build successfully. Alternatively, you could set a configuration to fire only if a certain file was updated in it's repository.
- **Inheriting project settings**. Configurations inherit settings from its project, however these can be overridden per configuration. **Permissions**, **Variables** and **Cleanup Policies** are all inherited automatically from the configuration's project, however these can be set per configuration.