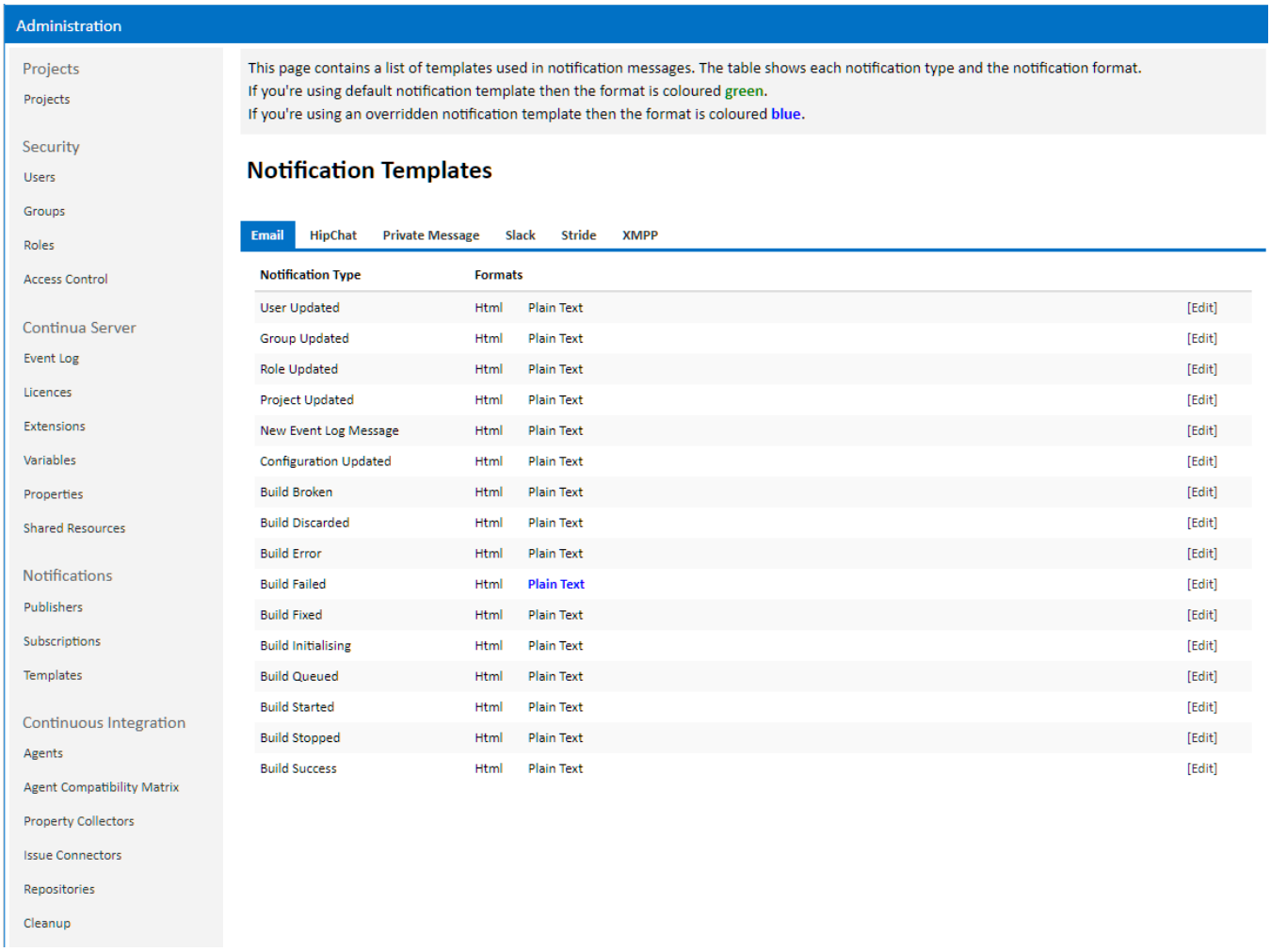
Templates

This page describes the features available from version 1.9 - see Templates (pre-1.9) for older versions

Notification templates allow you to control exactly what content is sent for each notification type. Each Publisher type has a set of template formats which can be edited provide content for each notification type.

Templates



Default template content is provided for each combination. When a template has been edited the format name is coloured blue.

Click the 'Edit' button for each notification type to edit the associated templates.

Editing Templates

Edit Email "Build Failed" Templates [List All Templates]

Html | Plain Text

```
Build '{{Version}}' of configuration '{{Project.Name}}:{{ConfigName}}' which was started by {{StartedBy}} has a status of {{Name}}
{{Project.Name}}:{{ConfigName}}
Status:   {{Name}}
Version: {{Version}}
Started By:   {{StartedBy}}

Repository  / Branch / User / Issues / Comment
{% for change in ChangeSets %}{{change.RepositoryName}} / {{change.Branch}} / {{change.RepositoryUserName}} / {% for issue in change.Issues %}{% if forloop.last == true
%}{{issue.Name}}{% else %}{{issue.Name}}, {% endif %}{% endfor %} / {{change.Comment}}
{% endfor %}
View Build : {{BuildUrl}}
```

The template content is shown in an editable text area under a tab for each template format. This uses DotLiquid markup syntax which allow you to specify variables using the {{variablename}} notation, control flow with {{for}} loops and {{if}} conditions and many others.

Editing the content will mark the associated tab with an asterisk



and buttons will appear for saving the templates.



Once an edited template has been saved, the tab will be marked with a pencil



and buttons will appear allowing you to revert the template to the default format.



Below the editable text area, is a list of variables that are available to be used in the template. These are specific to the notification type, so build-related notification types, for example, will give you access to the build's version, status, associated changesets etc. Variables can be inserted into the template using curly brackets e.g. {{Configuration.Name}}

Available Variables

Variable Name	Type	Description
Name	String	
NotificationType	NotificationEventType	
BuildUrl	String	Build URL
▶ Changesets	List(ChangesetDTO)	List of change sets
ConfigName	String	Name of configuration
ConfigurationUrl	String	Configuration URL
MessageType	PrivateMessageType	
▶ Project	EntityDTO	Build project
ProjectUrl	String	Project URL
RestrictToUsers	List(Guid)	
StartComment	String	Start comment
StartedBy	String	User who started the build
Status	String	Build status
StatusMessage	String	Build status messages
Subject	String	
TriggeringRepositoryId	Guid	Triggering Repository Id
▶ Variables	List(VariableDTO)	List of variables
Version	String	Version of build
UniqueId	Guid	Unique Identifier for Notification