



# #define 4 User Guide

Version 4.0

© CGI Group Inc.

**CGI**

Experience the commitment®

# Table of Contents

Section	Slides
Wiki	5-7
Tickets	8-12
Files	13-15
Planning	16-21
Cooperate	22-24
Settings	25-31
Agile Features	32-34
Desktop Applications	35-36



# User Interface

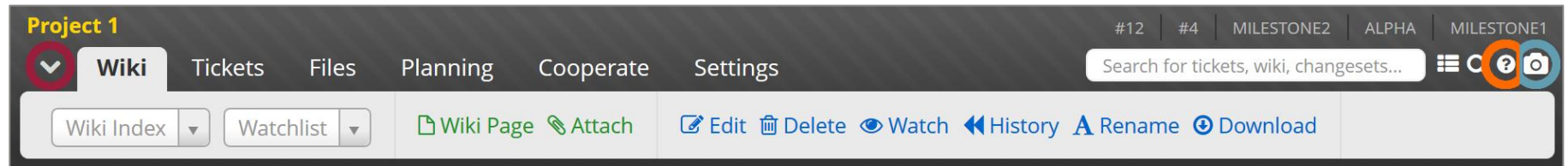


**CGI**

Experience the commitment®

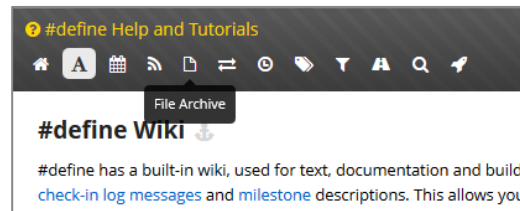
# User Interface Ribbon

The #define ribbon helps you navigate between components inside the user interface. Available on every page, the ribbon gives you a convenient and familiar way to access shortcuts, use major navigational links and search project content. Useful tools such as the quick-navigation menu, help guide and interactive tour are also available from the ribbon.



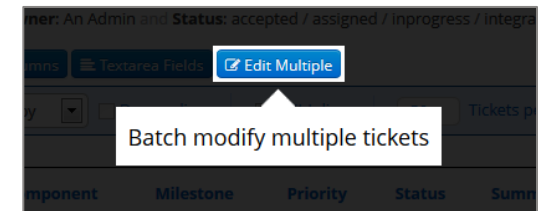
## Quick Navigation

Click the drop down arrow to open. Here you can create tickets, navigate to different projects and log in / out.



## #define Help Guide

Click the question mark icon to open. The guide includes a range of information and examples about the features and components available in #define.



## #define Tour

Click the camera icon to open. The interactive tour highlights the key features available on every page. Navigate either using the button at the bottom of the page or your keyboard.

# User Interface Wiki



**CGI**

Experience the commitment®

# User Interface – Wiki Overview

Wiki pages are a powerful tool which encourage collaboration, make the distribution of information easier and ensure your project documentation is centrally stored and accessible to all team members. The *WikiStart* page is equivalent to the project homepage. When you log into #define it is the first page you'll see.


**Welcome to your #define 4 project** 

The *WikiStart* page, like all wiki pages, is entirely customisable. Using the ribbon, you can easily navigate your project's pages, make changes to existing pages, as well as create new wiki pages. The watchlist feature allows you monitor specific pages within your project, and will automatically email you whenever changes are made to any of your watched pages.

Change auditing and version control is a core principle of #define. Whenever a wiki page is updated, it is given a new version number. By visiting a wiki page's history, you can easily revert to an older version of the page, or simply view what has changed between it's versions.

Wiki Index ▼

Watchlist ▼


 Wiki Page  Attach

 Edit

 Delete

 Watch

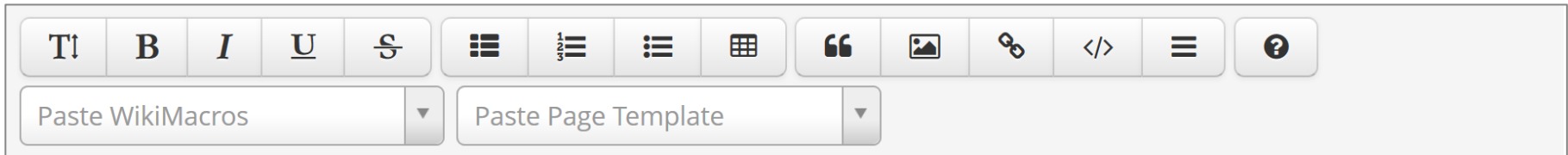
 History

 Rename

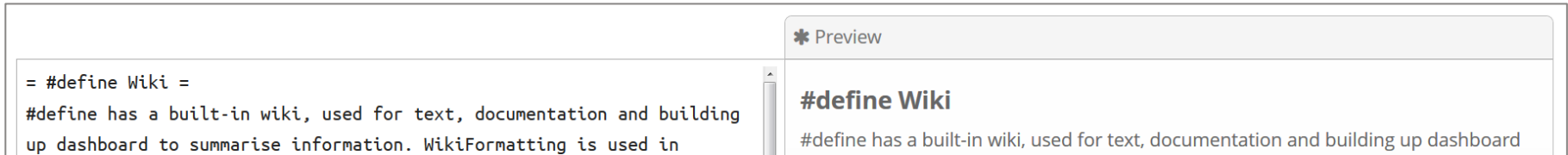
 Download

# User Interface – Wiki Editing

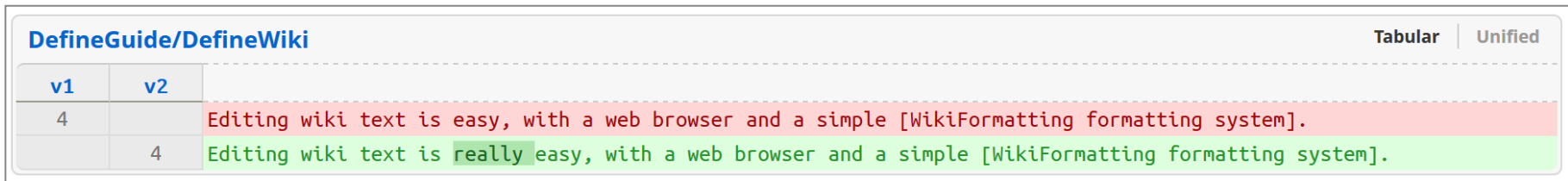
The wiki editor allows you to easily make changes to your wiki pages. The **toolbar** provides a simple interface to format text and insert other elements, such as tables and images. It also lists a number of **macros**, which allow you to insert dynamic information such as tickets matching a query, all users within the project, and the progress of milestones, within your pages.



The **preview** box – which can be alongside or beneath your content – shows how your changes will affect the page in real time



Before saving your changes, you may wish to first **review** them to make sure that it's exactly what you want.



# User Interface Tickets



**CGI**

Experience the commitment®



# User Interface – Tickets

## Query Page Overview

At the heart of #define is the ticket system, which allows users to document all the work to be completed by the project, and assign individual tasks to members of the team. When visiting the tickets section of #define you will first be greeted by the **query page**, from which you can easily keep track of your project's tickets. By applying additional **filters**, you can narrow down your results.

Filters Columns Textarea Fields Edit Multiple  
Clear All Filters

Filter	Rule	Values	Actions
Owner	is	An Admin	

The resulting tickets are **colour-coded** by *priority*, and you can change the **columns** by clicking the button shown in the image above.

Ticket	Component	Milestone	Priority	Status	Summary	Type
#4	component1	milestone4	blocker	assigned	<b>Nunc mauris. Morbi non sapien molestie orci tincidunt</b>	defect
#34	component2	milestone4	critical	assigned	<b>luctus et ultrices posuere cubilia</b>	defect
#49	component2	milestone4	major	assigned	<b>dictum placerat, augue. Sed molestie. Sed id risus quis diam</b>	defect
#10	component2	milestone1	minor	assigned	<b>elementum sem, vitae</b>	defect

# User Interface – Tickets

## Query Page Saved Queries

To avoid the stress of repeatedly adding the same complex filters to your query, #define includes the ability to **save your queries**. Simply click on the “Save these filters” button in the ribbon to be taken to a new page where you can save your custom query. You can then use the “Queries” dropdown (also in the ribbon) to navigate to previously saved queries. Project managers have the additional ability to create custom queries which are available to all members of the project.

### My Custom Query (13 matches)

**Using Filters:** **Owner:** Me and **Status:** accepted / assigned or **QA Contact:** You

The #define **default query** filters tickets which it assumes are most relevant to *you*. If this is not the behaviour you’d like, just click the “Set as default” button in the ribbon whilst viewing the query you’d like to become your default.

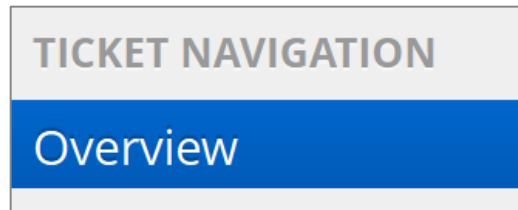
 Set as default

# User Interface – Tickets

## Ticket Page Overview

Tickets in #define are completely customisable. A project can have as many different types of tickets, containing as many fields, as they like. Consequently its important you familiarise yourself with the structure of ticket pages and the features available to you. All tickets however, have an **overview** section at the top of the page. This provides you with a summary of the ticket, and, like the query page, is *colour-coded* to show at a glance, the priority of the ticket.

Each ticket type uses a **workflow** which determines the life cycle of a ticket. Workflows can be particularly complex and require that users take specific actions in order to advance the state of a ticket. To view the workflow used by a specific, click on the “Workflow” button in the ribbon. This will open up a dialog which visualises the workflow, and can help clarify the work process.



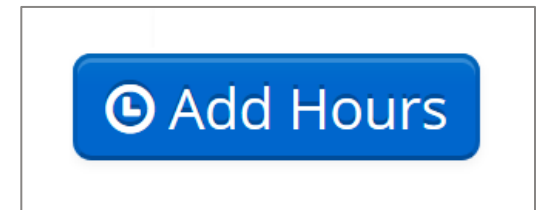
### Navigating Tickets

Use the ticket navigation box on the left hand side of the page to navigate to any section of the ticket with ease. The section highlighted in blue also represents where you current are in the ticket, so you never get lost!



### Modifying Fields

The ticket page is split into two sections – one, where the you can view the properties of the ticket, and another, where you can edit the properties (if you have the permissions). Whilst viewing a field, simply click “Edit” to jump to the editable section.



### Adding Hours

Keeping track of the time spent on individual tickets can be extremely important to a project (*agile* projects especially). The *Add Hours* button can be found just to the right of the ticket summary.

# User Interface – Tickets

## Ticket Page History


Every ticket has a History section, which displays a chronological log of all the changes made to the ticket. The section is split up into a number of tabs. The first tab, **Changes**, gives an overview of all developments to the ticket, and the others provide a more detailed explanation of the ticket state.

**Baker, James** 1h Link Edit Reply  
 **Milestone** changed from *Bravo* to *Alpha*

---

**Smith, Terry** 2h  
 **Cc** *Collins, Charlotte* added  
 **Depends on** #25 added

The **Linked Tickets** tab shows which other tickets are related to the current one, and *how* they're related. You can even link to tickets in other projects. *Remember to save your changes after linking tickets!* The **Subscriptions** tab shows a list of all users receiving email notifications for any changes made to the ticket. This generally includes the reporter of the issue, the Quality Assurance contact, and any other members who are CC'd in. The **Changesets** tab shows all code changes which have referenced the current ticket's ID in their commit message (using the “#” prefix).

 Changes (5)  Attachments (0)  Linked Tickets (1)  Subscriptions  Changesets

# User Interface Files



**CGI**

Experience the commitment®

# User Interface – Files

## File Browser

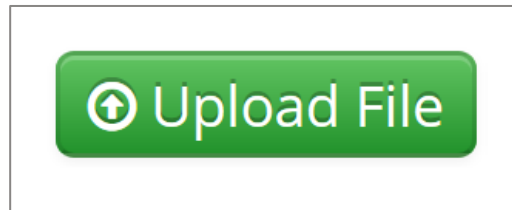
You can upload project documents and source code to the #define file repository. This facilitates and encourages collaboration, allowing team members who are geographically spread access to the same synchronised files. #define uses the version control system Subversion to retain copies of every document and file uploaded to its repository. It also collects information relating to any subsequent modifications of these files – allowing authorised users to review modifications, compare previous versions and revert changes if necessary.

Only users with the necessary permissions are able to modify files stored in the project repository – while #define records who made the change, when the change occurred, what the change involved and why the change was made. You can browse, upload, download and manage files via the #define web interface.



### Navigating the File Browser

Click on a folder's expander (triangle icon) to view its contents, without needing refresh the entire page. The “Jump to Path” and “View Revision” selections in the ribbon also allow you to quickly find what you want.



### Making Changes

Use the buttons at the top of the browser to perform a number of browser operations, including adding and removing folders, and more. To perform actions on individual files and folders, click the chevron icon next to their name to bring up the context menu.



### Integrating with Subversion

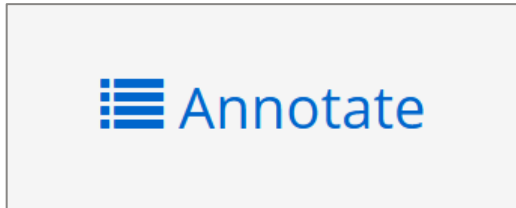
With the #define browser it's simple to integrate with any Subversion client (discussed later). Just click on the link in the ribbon from any browser page, and then copy the link from the dialog to get started. *New – Git support coming early 2014!*

# User Interface – Files

## Viewing Individual Files and Changesets

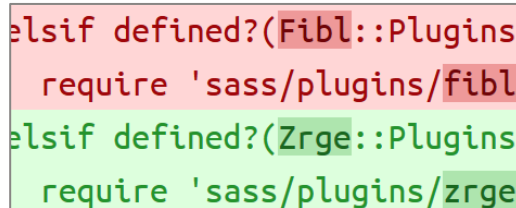
A particularly useful feature of the browser is the online source browser, which has syntax highlighting capabilities to make source code more readable and user friendly.

50	<code># @see #update_stylesheets</code>
51	<code>def check_for_updates</code>
52	<code>  return unless !Sass::Plugin.checked_for_updates   </code>
53	<code>    Sass::Plugin.options[:always_update]    Sass::Plugin.options[:always_check]</code>



### File Annotations

Click the “annotate” link in the ribbon to view a file with additional information about when each line was added (also known as a *blame* view). Clicking on the changeset link beside a line will load a popup explaining who last changed the line, and why.



### Colour-Coded Changesets

Changesets are colour coded, so that it's easy to see what has been added and removed. In addition, you can customise the view, by altering such things as the number of lines visible around the changes, viewing the versions side-by-side, and ignoring blank lines.



### Revision Log

Find the “Revision Log” for any file or folder in the ribbon. It provides a comprehensive list of all the changes made to that particular file or folder, documenting what and when the change was, who it was modified by, and the message provided with the commit.

# User Interface Planning



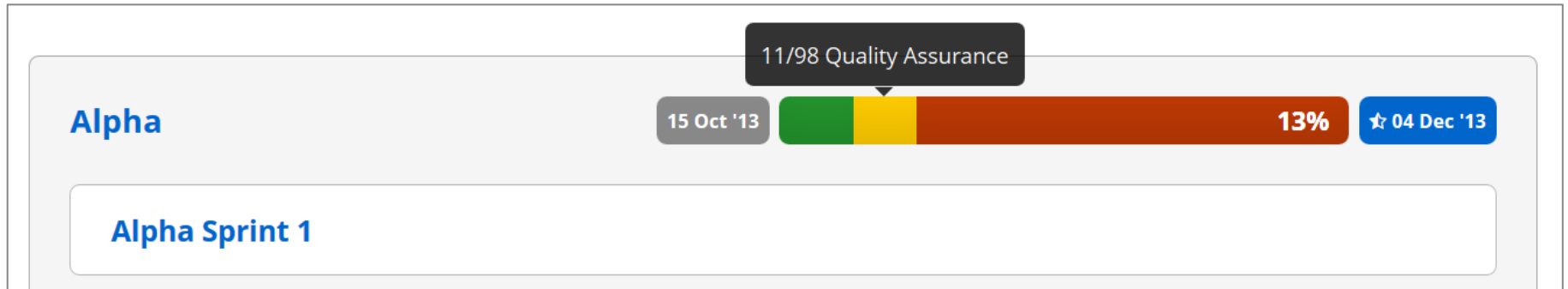
**CGI**

Experience the commitment®



# User Interface – Planning Roadmap

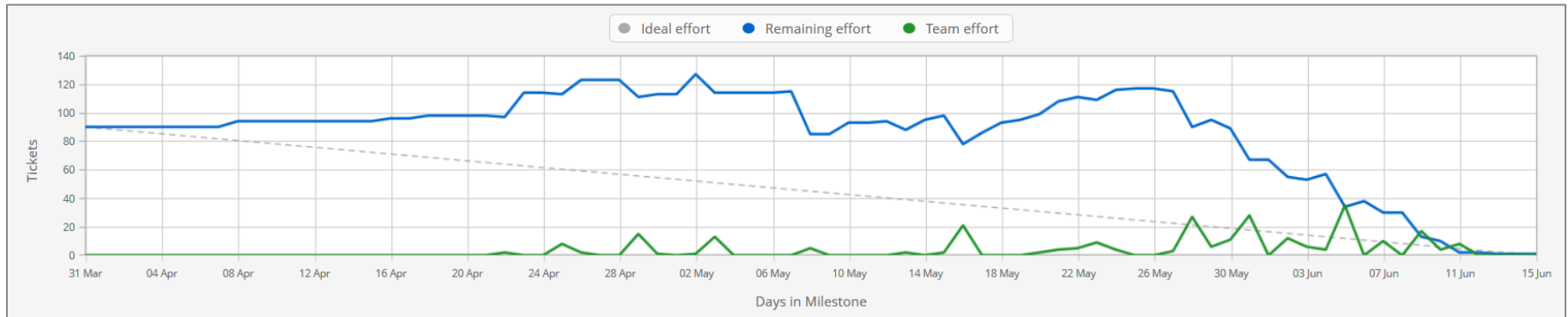
The roadmap page provides an overview of all milestones within a project and their progress. A progress bar is shown for each milestone which is split into 3 categories: **Completed**, **Quality Assurance** and **Open**. By hovering over any one of these sections you can see exactly how many tickets are in that state, and clicking the section will take you to a query page with the relevant filter. The scheduled start and completion dates are to the left and right of the progress bar respectively, and are colour coded. If a milestone is overdue, the end date label will change to red to signify that it needs to be addressed.



The milestone structure is a **hierarchy**, where top level milestones can have child milestones, to work on a subset of tasks. In *Agile* development, this structure is most commonly referred to as *Releases and Sprints*, where sprints usually last around a week. By default, **completed milestones** are hidden from the interface, but this can be edited by clicking on the “filters” button at the top of the roadmap page.

# User Interface – Planning Milestones Page

The milestone page provides additional information for each milestone. The most significant of these is perhaps the **Burn Down chart**, which maps how the amount of tickets, hours, or story points, has developed over time. To change the *units* of the chart, or to *print* it out, click the links in the ribbon. **Note:** a milestone needs a start and a due date in order for a burn down chart to be generated.



Additionally, the milestone page allows you to **break down analysis** of development by different categories. For example, you can see which components have had the most work against, and how the status of development is different across components. Simply select the field you wish to sort by in the “Analysis by field” section to change.

User Interface

8 / 46

Database

5 / 52

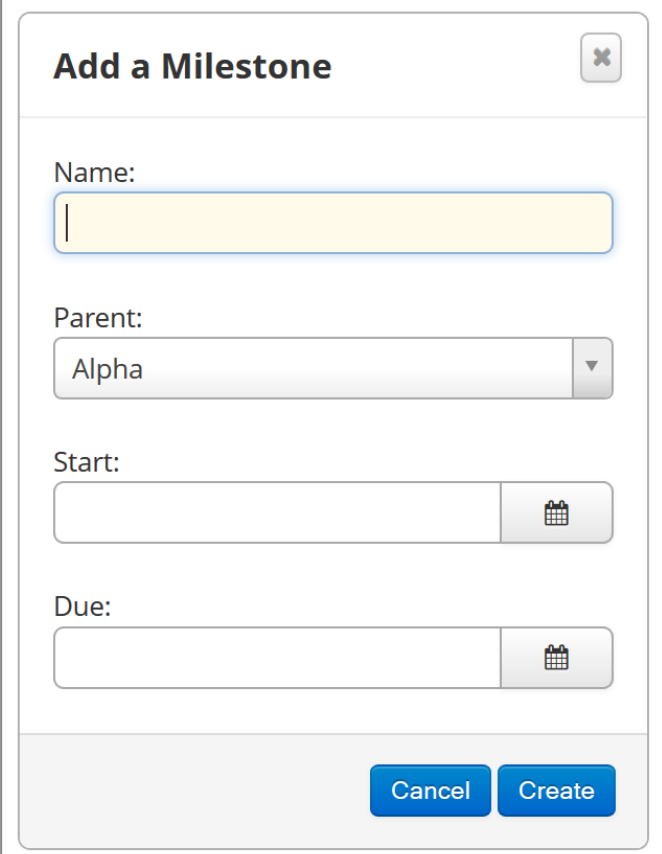
# User Interface - Planning Milestone Management

The easiest way to **add**, and **edit** individual milestones is through the *Planning* section. To add a new milestone, simply click the “Milestone” option in the *Create* (green) section of the ribbon.

This will cause a dialog to pop-up from which you can create your milestone (right). The dialog allows you to add milestones in one click, but if you want to add more information to a milestone after you’ve created it – you can add a **description** to it by clicking “Edit” in the ribbon from any milestone’s page.

The edit page also allows you to set a milestone as **completed**. By checking the *Complete At* box, the date picker to the right of it will become enabled, and allow you to specify exactly when the milestone was completed. It’ll also enable the option to move any **open tickets** to another milestone.


If you have the permissions, you may also want to administrate **multiple milestones** at once through the **admin section**. You can get to this by visiting the “Settings” section in the ribbon, and then finding “Milestones” in the “Administration” ribbon, also in the ribbon. **Note:** *The administration drop down, and all dropdowns of similar design, can be filtered using your keyboard.* On the *Manage Milestones* page you can **delete multiple milestones** in one go, as well as select a **default milestone** (to be used in the tickets and other pages e.g.)




**Add a Milestone** ✕

Name:

Parent:  ▾


Start:  

Due:  

# User Interface – Planning Timeline

The timeline gives a overview of all the changes that have been made to your project in a single report. It lists **all events** that have happened in chronological order, displaying a brief description of each event and the person responsible for the change. You can also **download** the timeline as an RSS Feed through the link in the ribbon.


 13:29 Changeset [8] by Jones, Samuel

 trunk/bin/pylint

Adding pylint folder

You can **filter** your results based on the date, type of event and author. You can search for particular events, such as wiki changes, file archive changesets, or mailing list messages. ***Note:** Remember to include a milestone in your filter if you want to search for ticket or milestone updates.*

## Change View of Timeline

View changes from  

# User Interface – Planning

## Manage Hours

The manage hours home page closely resembles the design of the standard Tickets **Query page**. You can add **filters** and **columns** to help narrow down your results.

#175	<a href="#">Quality Assurance</a>	7.00	2013-11-30	<a href="#">1.0</a>	<a href="#">domain\wiley</a>
#124	<a href="#">Researching possible solutions</a>	10.00	2013-12-01	<a href="#">1.0</a>	<a href="#">domain\peck</a>
#49	<a href="#">Merging with release branch</a>	6.00	2013-12-03	<a href="#">2.0</a>	<a href="#">domain\hart</a>
#103	<a href="#">Development</a>	3.00	2013-12-02	<a href="#">2.0</a>	<a href="#">domain\berg</a>

Click on the **Hours by user** link in the ribbon to see a breakdown of hour many hours each member of your project has logged. You can filter these results by date and milestone. Click on any of the worker's names to see exactly what they've logged hours against and why.

 [Hours by user](#)

### Hours by User

View from

2013-11-28

to

2013-12-31

Milestone

Alpha Sprint 1



 Update

# User Interface Cooperate









**CGI**

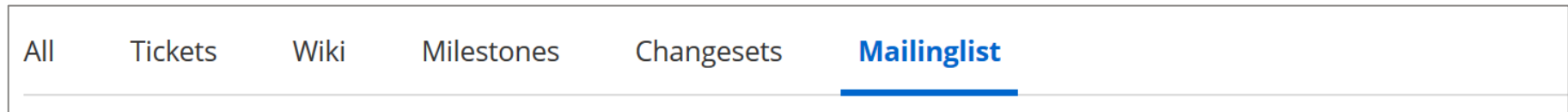
Experience the commitment®

# User Interface – Cooperate Mailing Lists Overview

In #define you can create mailing lists which users in your project can **subscribe** to. In addition to the obvious benefits of making information more easily available to others and encouraging collaboration, all mailing list messages are **archived** and **searchable** inside the #define 4 user interface. All mailing list addresses follow the schema of [name@projectid.define.logica.com](#). For more information on the project, including which members are currently subscribed to it, click on the “More info” link. Click on the name of the project to open up an archive of the list’s conversations.

Name	Description	Members	Conversations	Actions
<a href="#">General</a>	General project mailing list	141	0	 <a href="#">Send mail</a>  <a href="#">Unsubscribe</a>  <a href="#">More info</a>
<a href="#">Managers</a>	Project managers mailing list	71	0	 <a href="#">Send mail</a>  <a href="#">Unsubscribe</a>  <a href="#">More info</a>

To **search** for mailing list conversations, use the #define search (located in the ribbon), and filter for mailing lists



To create a **new mailing list**, select the ‘Mailing List’ icon in the *Create* section of the ribbon.



# User Interface – Cooperate Mailing Lists Management

Once you have created a mailing list, you may want to add users or group subscriptions, or modify the access and permission settings. To do this navigate to the **Mailing List administration page**, which can be found by visiting the “Settings” section of #define, clicking on the “Administration” dropdown, and locating “Lists”.

From this page, you can fine tune the features of individual lists by clicking on their name in the table. From these pages you setup who is able to **access** and **post** to the group, and who replies to the list are sent to.

Access:	Public: Viewable by people who can view this project
Posting restricted to:	Members
Replies to go:	List

You can also use this page to add both individual member and group **subscriptions**. The individual user subscription table shows each member, and explains *how* they’re subscribed – if their subscription is through being a member of a group or just an individual case. It also lists if user has declined subscription to the group (by unsubscribing through the standard mailing list page)

Project Viewers	+ Add Group	domain\miles <miles@domain.c>	+ Add User
-----------------	-------------	-------------------------------	------------



# User Interface Settings



**CGI**

Experience the commitment®

# User Interface – Settings

## Administration – Components

Tickets which relate to the same feature or area of your system can be grouped using the component field. They can then be quickly searched using the ticket filter functionality. You can manage the component settings in your #define project by clicking on the “Administration” dropdown and choosing “Components” which is part of the “Ticket System” section. To **edit** a component, and to give a component a description, simply click on it’s name from the *manage* page.

	Name	Owner	Default	Disabled
<input type="checkbox"/>	Database	domain\haney	<input type="radio"/>	<input type="checkbox"/>
<input type="checkbox"/>	User Interface	domain\acosta	<input type="radio"/>	<input type="checkbox"/>

To **add** a new component, simply fill out the form at the top of the page. The mandatory **owner** attribute specifies who the ticket should be assigned to by default if no owner is provided.

### Add Component

Name:

Owner:

# User Interface – Settings

## Administration – Priority and Versions

**Note:** Both of these pages are largely similar to the components page described in the previous slide. The importance of an issue can be described by its **priority**. By default #define has the priority values *blocker*, *critical*, *major*, *minor* and *trivial*. When you add and edit priorities remember to assign an appropriate *order*, as this is what is used to *colour-code* tickets in the Query page etc.

	Name	Owner	Default	Disabled
<input type="checkbox"/>	Database	domain\haney	<input type="radio"/>	<input type="checkbox"/>
<input type="checkbox"/>	User Interface	domain\acosta	<input type="radio"/>	<input type="checkbox"/>

You can use the **version** field to organise work against particular versions of your project. Each version also has an associated release date. Like the priority field you can select a default value.

# User Interface – Settings

## Administration – Custom Fields

Using custom fields you can project specific properties to your tickets. These fields can take a variety of input types and be assigned a default value if appropriate. You can also perform queries based on custom field values on the ticket query page.

**Add field**

Internal field name:	<input type="text" value="a-z letters and numbers only"/>
Show as:	<input type="text" value="Text"/>
Treat as:	<input type="text" value="String"/>
English label:	<input type="text" value="User friendly name to show on forms"/>
Default value:	<input type="text"/>
Format:	<input type="text" value="Plain"/>

**Note:** After adding a custom field to your project you will need to use the Ticket Type Layout editor to add the field to a specific ticket type. After visiting the layout page, you'll notice the new field in the "Unused Fields" box.

# User Interface – Settings

## Administration – Custom Fields Types

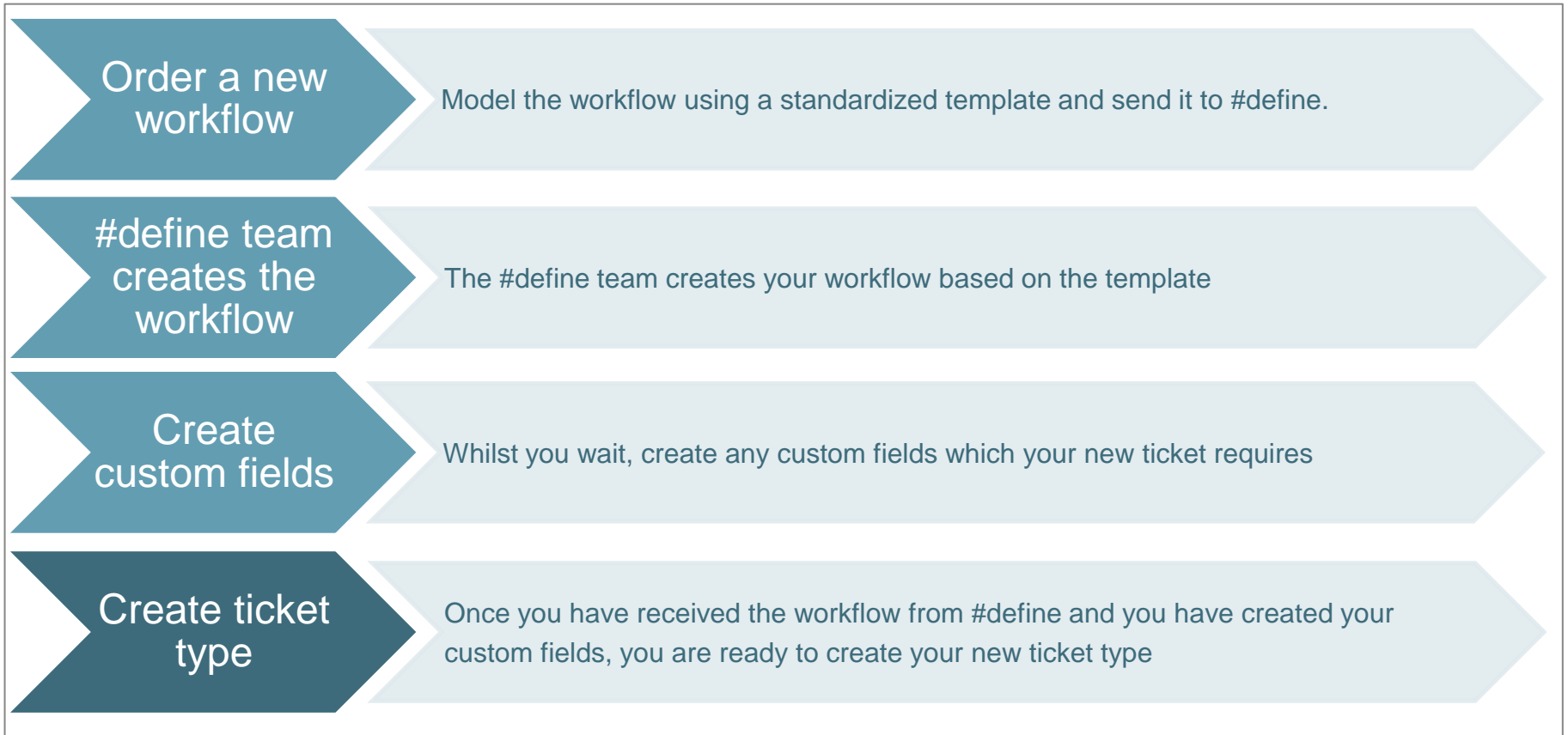
The following table describes the various types of custom fields.

Type	Additional Properties	Usage
Text	<b>Format</b> – Either <i>Plain</i> or <i>Wiki</i> , which adds a toolbar to the field and allows for additional formatting.	A plain text box, without any limitations on the contents.
Textarea		Similar to <i>Text</i> but allows for multiple lines of information to be added.
Select	<b>Options</b> – Separated by a new line (an initial blank line makes the field optional)	Limited value dropdown list.
Radio		Limited value radio buttons. User can select a single value.
Checkbox		Limited value checkboxes. User can select multiple values
Date		Adds a text box with a date picker
User List	<b>Included Groups</b> – Either select “All” or manually select individual groups to have their <i>members</i> included in the dropdown. <b>Note:</b> only groups marked as “Show in user select areas” in the Access and Groups section of the site will be visible.	Creates a dropdown containing all members of the selected groups. The groups can include remote groups (ACC), which will be loaded in whenever a user clicks to expand the group in the dropdown.

# User Interface – Settings

## Administration – Managing Workflows and Ticket Types

The following table describes the various types of custom fields.



# User Interface – Settings

## Administration – Ticket Types

To view and manage all of your project's ticket types, visit the *Types* page which can be found in the *Ticket System* section of the *Administration* dropdown. Each ticket type has a number of properties which you can change by clicking the links in the “Actions” column.

enhancement	default (define-default 0.4.4)	<a href="#">Permissions</a>	<a href="#">Layout</a>	<a href="#">Workflow</a>	<a href="#">Export</a>	<a href="#">Delete</a>
story	define-simple 0.1	<a href="#">Permissions</a>	<a href="#">Layout</a>	<a href="#">Workflow</a>	<a href="#">Export</a>	<a href="#">Delete</a>

The **Permissions** section for a given ticket type allows you to specify exactly which groups are able to view and make changes to tickets of the type. **Note:** You may find that some checkboxes are both checked and disabled, this is because this user group has been granted this permission for all ticket types.

The **Layout** section allows you to completely customise which fields are included in your ticket type, where they appear on the page, and who can edit them. Simply drag the fields and columns around to reorder them. You can also split certain sections up into tabs. **Note:** actual column count depends on the device (e.g. mobiles only have 1), so set a colspan of 4 to make a field always fixed width

### Category

#### Component

component (select / string)

Colspan: 1

New Line:

Required:

Always editable:

Annotation/help text

#### Milestone

milestone (select / string)

Colspan: 1

New Line:

Required:

Always editable:

Annotation/help text

#### Version

version (select / string)

Colspan: 1

New Line:

Required:

Always editable:

Annotation/help text

### Unused Items

#### Fields

Story Points  
effort (text plain / float)

#### Groups

# User Interface Agile Features



**CGI**

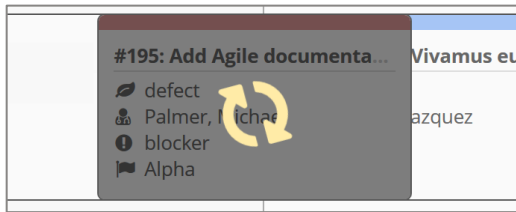
Experience the commitment®



# User Interface – Agile Features

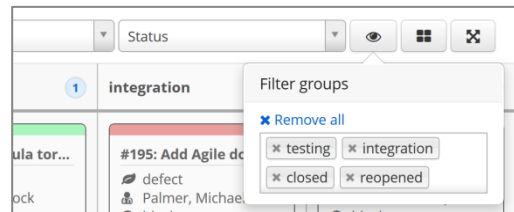
## Task Board

The #define task board can be used to manage and keep up to date with the progress of your project's milestones. The page is kept up to date, and any developments within the project are reflected in real time. Traditionally, task boards are used to assist *agile* development, but the #define task board can be used by all projects who want to organise tickets: by owner, by status, or by any other fixed-value field used by your project.



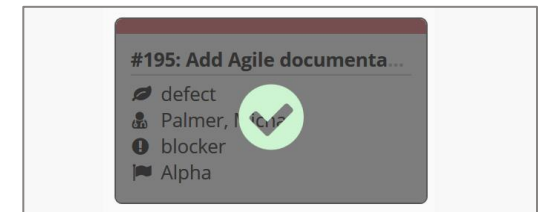
### Always up to date

The #define task board updates automatically, so whenever a change is made to your project, and the ticket was visible on your task board, the ticket will be temporarily covered by a yellow reloading icon, which will disappear after a few seconds. Tickets moved both in and out of the scope of your current filter will also be shown. This feature means that the task board can be useful for both standup meetings, and to make visible to the members of your project at all times.



### Customising Appearance

Use the filters above the task board to customise your appearance of the board. The two dropdowns allow you to change which milestone to show tickets for, and how to group the tickets. You can also filter the visible groups, which is usually required when grouping by owner or reporter for example. There's also the option of a condensed view, and you even make the board fullscreen, which might be useful if you want the board to be always available for your project to see.



### Making Changes

To change the state of a ticket, simply drag and drop it into the desired group. The ticket type's workflow is still respected, and so any invalid moves are highlighted in red. Once completed, the ticket will attempt to save. If successful, a green box will appear, otherwise you'll see a red exclamation sign. Click on it to be given a detailed description of why the ticket failed to save. More often than not a ticket will fail to save because of a lack of permissions.

# User Interface – Agile Features

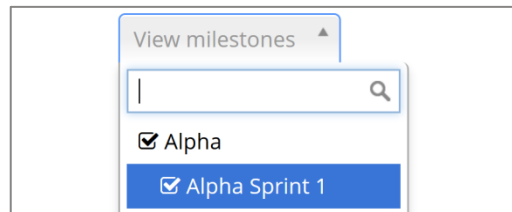
## Backlog

The #define backlog allows you organise the state of your tickets, by reprioritising individual tickets, and by moving tickets into milestones. For *Agile* development teams this process should be carried out regularly, ensuring that the most important work is always being completed. The *Product Backlog* represents all tickets which are currently not scheduled for any milestone. Ideally, project's should regularly trim the product backlog, and close tickets should if it's unlikely the team will never get around to completing them.

#167	in faucibus orci luctus et ultrices posuere cubilia Curae;	8h
#57	cursus et, magna. Praesent interdum ligula eu enim. Etiam imperdiet	🌀
#189	montes, nascetur ridiculus mus. Proin	9h
#195	Add Agile documentation	3h

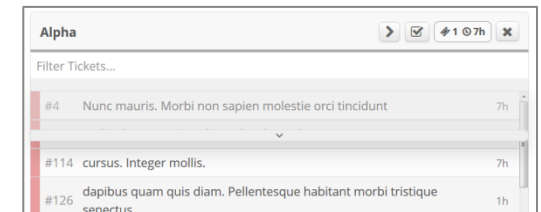
### Reordering Tickets

By default, tickets will be ordered by their “priority” (e.g. blocker, critical). However, you may want to be more specific about which tickets should be completed first. To reorder an individual ticket, simply drag and drop it into it's new position. You'll see a loading icon will appear momentarily whilst this new position is saved, and will disappear soon after. Similar to the task board, if an error appears, click on the exclamation mark to be given a detailed explanation of why.



### Customising Appearance

Use the *View milestones* dropdown at the top of the page to add and remove up to 3 milestones (excluding the product backlog). Once you add a milestone, it will appear in the backlog, and load it's tickets. This process might take a while, particularly if a milestone has many hundreds of tickets. You can then reorder the milestone boxes by clicking and dragging the milestone's title bars.



### Moving to a milestone

For an individual ticket, simply drag and drop it into the new milestone. You can also move *multiple* tickets into another milestone. To do this, click on the grey bar below the “Filter tickets” box, and drag it down to select as many tickets as you wish. This will update the ticket and hours count box in the milestone header. Then, click the *right chevron* icon in the header to move all the tickets across to the milestone to the right. You can also select all tickets in the header.

# Desktop Applications



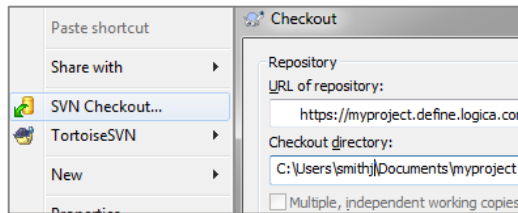
**CGI**

Experience the commitment®

# Desktop Applications

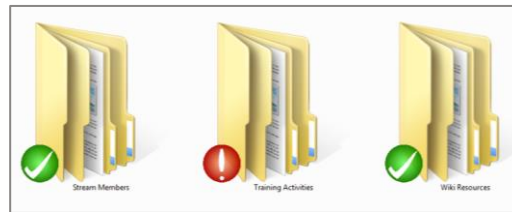
## Subversion Client

You can also access and manage the file repository from a subversion client of your choice. #define recommends **Tortoise SVN** (<http://tortoisesvn.net>), which is a free, open source GUI client that integrates into the Windows Explorer shell.



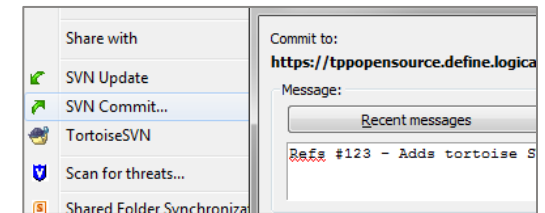
### Check Out

To download copies of files stored in the #define, use windows explorer to navigate to a directory of your choice, right click and select *SVN Checkout*. A dialogue will then prompt you to provide the URL of the file or folder you wish you copy, which you can get from the #define browser (see slides).



### Make Changes

You can make changes to your local (working) copies without modifying the original versions. TortoiseSVN automatically adds icons to each file and folder to signify if it has been changed or not.



### Commit

Once you're happy with your changes, you can upload them back to the #define central repository. Right click anywhere within the windows explorer whilst browsing your working copy, and select commit. From here you can see exactly what you've modified and add a commit message.

## Our commitment to you

We approach every engagement with one objective in mind: to help clients succeed

