AllChange Getting Started Guide

AllChange Getting Started Guide

Version 10.0,Sep 2014

Email: support@intasoft.net
Web: http://www.intasoft.net

DISCLAIMER

While every effort is made to ensure accuracy, Intasoft Limited cannot be held responsible for errors or omissions, and reserve the right to revise this document without notice.

COPYRIGHT

TRADEMARKS

ACKNOWLEDGEMENTS

Table Of Contents

Getting Started

Welcome to ***AllChange***, the ultimate in flexible change control and configuration management tools. All-Change provides power with flexibility to ensure that you have the level of control that *you* want.

AllChange is supplied with several configurations to illustrate different possible processes and procedures, but you can tailor/configure most aspects of the system to suit your particular environment and requirements.

AllChange is a very comprehensive tool with a plethora of features and capabilities, this guide is designed to help you get started with using one of the supplied out-of-the-box configurations and to provide pointers to more detailed information and information about tailoring the configuration to suit your requirements.

Following this guide in the sequence below will take you through the steps needed to install, set up and perform some basic change management functions:

- **Step 1: Installation** - provides guidance on how to install the software
- **Step 2: Setting up your First Project** - provides guidance on how to set up your ***AllChange*** project (database and initial configuration)
- **Step 3: Setting up your Configuration Item Structure** - introduces the configuration management feature and how to get started with configuration items
- **Step 4: Creating your first Change Request** - introduces the change management feature
- **Step 5: Change Requests** - provides more information about the change management facilities
- **Step 6: Storing some files under *AllChange* Control** - provides guidance on how import your first files into ***AllChange***
- **Step 7: Parts** - provides more information about the configuration items
- **Step 8: Changing a file** - introduces the basic mechanism for changing a file under version control
- **Step 9: Creating Parts with no file** - introduces the facilities for modeling configuration items for which there is no file, e.g. for manufacturing or asset management

Installation

Having purchased ***AllChange*** or obtained an evaluation copy you will either have a CD or have a download from the Intasoft website.

Note that prior to installation you should have a Serial Number and License code supplied to you by Intasoft. If you do not have this information you will need to request it from info@intasoft.net

You will need Administrator permissions on the machine on which you are installing.

- To install from a download, simply run the executable file you have downloaded.
- To install from CD, insert the CD and the installation program should start immediately, if it does not then run the setup.exe program on the CD.

The installation will present you with a number of screens:

1. **Welcome Screen**: just select **Continue**
2. **Installation Type**: initially you should select **Full**, this will copy all files from media to your computer. After this has been performed the media will no longer be required.
3. **Intasoft License Agreement**, if you agree with the license agreement then tick **I agree**.
4. **SQL Express Installation**: ***AllChange*** requires SQL Server for its database. You may use your own existing SQL Server if you wish or you may use SQL Express. If using SQL Express you may use that supplied by Intasoft. Selecting **Install SQL Express** will install a SQL Express ***AllChange*** instance. *Note that SQL Express 2008 R2 requires .NET 3.5 before it can be installed.* See also SQL Server Installation in the Administrator Manual.
5. **Installation Directory**: specify here where you wish ***AllChange*** to be installed to. The default is `c:\acsys`, but you may choose any location you like. For evaluation purposes, local to your machines' C drive may be fine. If you have purchased ***AllChange*** and are installing for live usage then you will want to install it on a server which has enough space to hold all the data you wish to put into ***AllChange***, the biggest component of this will be the files that you wish ***AllChange*** to control. If installing onto a server we would recommend that you take the following steps prior to progressing the installation:
    - On the server create a folder for ***AllChange***, e.g. `\\myserver\AllChange`. This folder will hold the installation of ***AllChange*** and the ***AllChange*** data in separate subfolders (e.g. `\\myserver\AllChange\ACSYS` and `\\myserver\AllChange\AllChangeData`)
    - Export this folder as a share ensuring that all users of ***AllChange*** will have access to this share

    If installing on a server the Installation directory should then be entered as \\myserver\AllChange\ACSYS or similar.
6. **Product License**: you should enter the Serial Number and License code that has been provided to you. In the licensed users section the user you are currently logged on as will be shown there by default. Only users in this list will be able to use ***AllChange*** when it has been installed. More users can be added after installation, but it is important that at installation at least one user who will be administering ***AllChange*** is entered here. This should be the users Windows logon ID.
7. **Installation Options (1)**: this allows you select what you want installed from the media. Select **AllChange Program Files** and any other interfaces that users of AllChange may need. Any which are not selected will not be available to any users. Note however that further options may be installed at a later date by re-running the full install from media.
    - Microsoft Source Code Control Interface- this allows files to be checked in and out of development environments which support this interface, e.g. Visual Studio
    - Explorer Interface - this allows various AllChange file operations to be invoked from Windows Explorer, e.g. check in and out files.

  - Word/Excel Interface - this allows Word and Excel files to be checked in and out directly from within these tools and also allows these documents to be stamped with information from the AllChange database
  - Eclipse Interface- this allows files to be checked in and out directly from the eclipse development environment
  - Web Interface- this allows AllChange to be accessed via a web browser interface (rather than the default Windows Interface)

  On Continue from this step the files will be copied from the media to the chosen Installation directory.

8. **SQL Client Components Installation**: this will be shown if the installation detects that you do not have the necessary SQL Server client. Select **Install SQL Client Components** to install this.
9. **SQL Command Line Utilities**: this will be shown if you do not have SQL Server installed on this machine and the required SQL tools/command line utilities cannot be found. Select **Install SQL Command Line Utilities** to install these. See also SQL Server Installation in the Administrator Manual.
10. **Installation Options (2)**: this allows you to select which options are required for this particular workstation and what program group to create shortcuts in (i.e. what location from the Start | All Programs menu). Only those options which were selected in Installation Options (1) will be available here. Any options not selected at this time may be chosen at a later date by re-running the installation for a Workstation installation.
11. **Installation Options (3)**: if installing across a network (i.e. not onto the local machine) then this screen allows you to select whether to install the ***AllChange*** programs (executables) and help files locally or not. Executables stored locally will mean faster usage for the user, however more maintenance when there are upgrades. If executables are stored locally then so is the dictionary which will further improve startup time. The local dictionary is used if there is not a project or template specific dictionary. If help files are not local then you will need to make modifications to the registry to make the help files work. This may be done by the installation program if you wish. If you choose to do nothing with the help files, this means that they will reside on the server and may not function correctly (this is a Microsoft Windows limitation on accessing help files across a network). We would strongly recommend that at a minimum the help files are stored locally. You may choose any local directory you like to place these files.
12. **Word/Excel Integration**: if Word/Excel Integration was selected in Installation Options (2) then this screen will confirm the Word version and various locations which are needed by the integration. Files will be copied into these locations. If these appear blank, then AllChange installation was unable to ascertain the information from your registry. You will need to establish what these directories are from Word/Excel and enter them yourself.
13. **Eclipse Integration**: if the Eclipse Integration was selected during Installation Options (2) then this screen will request the location of your Eclipse plugins directory, the default is `c:\eclipse\plugins`, but this may not be correct for your eclipse integration.
14. **Setup Successful**: this is the final screen and indicates that ***AllChange*** has installed successfully

If you have only installed locally for evaluation purposes then ***AllChange*** will only be available to you on your machine.

If you have installed on a server and you wish to run ***AllChange*** in client/server mode (recommended), you will now need to perform a **Server Only** install, full instructions for this can be found in Setting up Client/Server Summary in the Administrator Manual.

If you installed the software on a server you will now need to perform a workstation install on any workstations that require access to ***AllChange***. More of these can be performed as an when required and you will initially probably only want to make it available to a small number of people whilst it is being set up and configured for use by your organisation. This may be only you, if this is the case and you performed the Full install above from your normal workstation, then you will not need to do anything further. If you installed it from the server machine itself, then you will need to perform a workstation installation on your normal workstation.

Having installed the software you now need to Create your First Project.

Workstation Installation

Having installed ***AllChange*** onto a server you will then need to perform a workstation installation onto each PC which is to be used to access ***AllChange***. This will copy files required locally on the PC from the location installed on the sever to the machine and set up required shortcuts and registry entries.

In order to perform a workstation installation you will need Administrator permissions on the machine on which you are installing.

Note that no media is required for the workstation installation.

Run the Setup.exe program in the SETUP subdirectory of where ***AllChange*** is installed on the server, e.g. `\\myserver\AllChange\ACSYS\SETUP\SETUP.EXE`

The installation will present you with a number of screens:

1. **Welcome Screen**: just select **Continue**
2. **Installation Type**: you should select **Workstation only (User)** or **Workstation only (Administrator)** as appropriate, depending on whether you are a normal ***AllChange*** user or an ***AllChange*** Administrator**.**
3. **Installation Directory**: specify here where ***AllChange*** is installed, this should default to the directory from which you have run the setup program (in the example above this would default to `\\myserver\AllChange\ACSYS`). This default value should be the correct so you should not need to change it.
4. **SQL Client Components Installation**: this will be shown if the installation detects that you do not have the necessary SQL Server client. Select **Install SQL Client Components** to install this.
5. **SQL Command Line Utilities**: this will be shown for a **Workstation only (Administrator)** installation only. If you do not have SQL Server installed on this machine and the required SQL tools/command line utilities cannot be found. Select **Install SQL Command Line Utilities** to install these. See also SQL Server Installation in the Administrator Manual.
6. **Installation Options (2)**: this allows you to select which options are required for this particular workstation and what program group to create them in (i.e. what location from the Start | All Programs menu). Only those options which were installed from the media during the Full install will be available here. Any options not selected at this time may be chosen at a later date by re-running the installation for a Workstation installation.
7. **Installation Options (3)**: if installing across a network (i.e. not onto the local machine) then this screen allows you to select whether to install the ***AllChange*** programs (executables) and help files locally or not. Executables stored locally will mean faster usage for the user, however more maintenance when there are upgrades. If executables are stored locally then so is the dictionary which will further improve startup time. The local dictionary is used if there is not a project or template specific dictionary. If help files are not local then you will need to make modifications to the registry to make the help files work. This may be done by the installation program if you wish. If you choose to do nothing with the help files, this means that they will reside on the server and may not function correctly (this is a Microsoft Windows limitation on accessing help files across a network). We would strongly recommend that at a minimum the help files are stored locally. You may choose any local directory you like to place these files.
8. **Word/Excel Integration**: if Word/Excel Integration was selected in Installation Options (2) then this screen will confirm the Word version and various locations which are needed by the integration. Files will be copied into these locations. If these appear blank, then AllChange installation was unable to ascertain the information from your registry. You will need to establish what these directories are from Word/Excel and enter them yourself. It will first attempt to install for all users, if this is not possible it will then install for the current user.

9. **Eclipse Integration**: if the Eclipse Integration was selected during Installation Options (2) then this screen will request the location of your Eclipse plugins directory, the default is `c:\eclipse\-plugins`, but this may not be correct for your eclipse integration.
10. **Setup Successful**: this is the final screen and indicates that ***AllChange*** has installed successfully

Setting up your First Project

Creating your First Project

Having installed ***AllChange***, before you can go any further you need to set up an ***AllChange*** project. An ***AllChange*** project does not necessarily bare any relationship to one of your organisations projects, it is an ***AllChange*** database together with configuration data. You may well only have one *live* ***AllChange*** project holding all the change management, configuration item and baseline artefacts for your organisation.

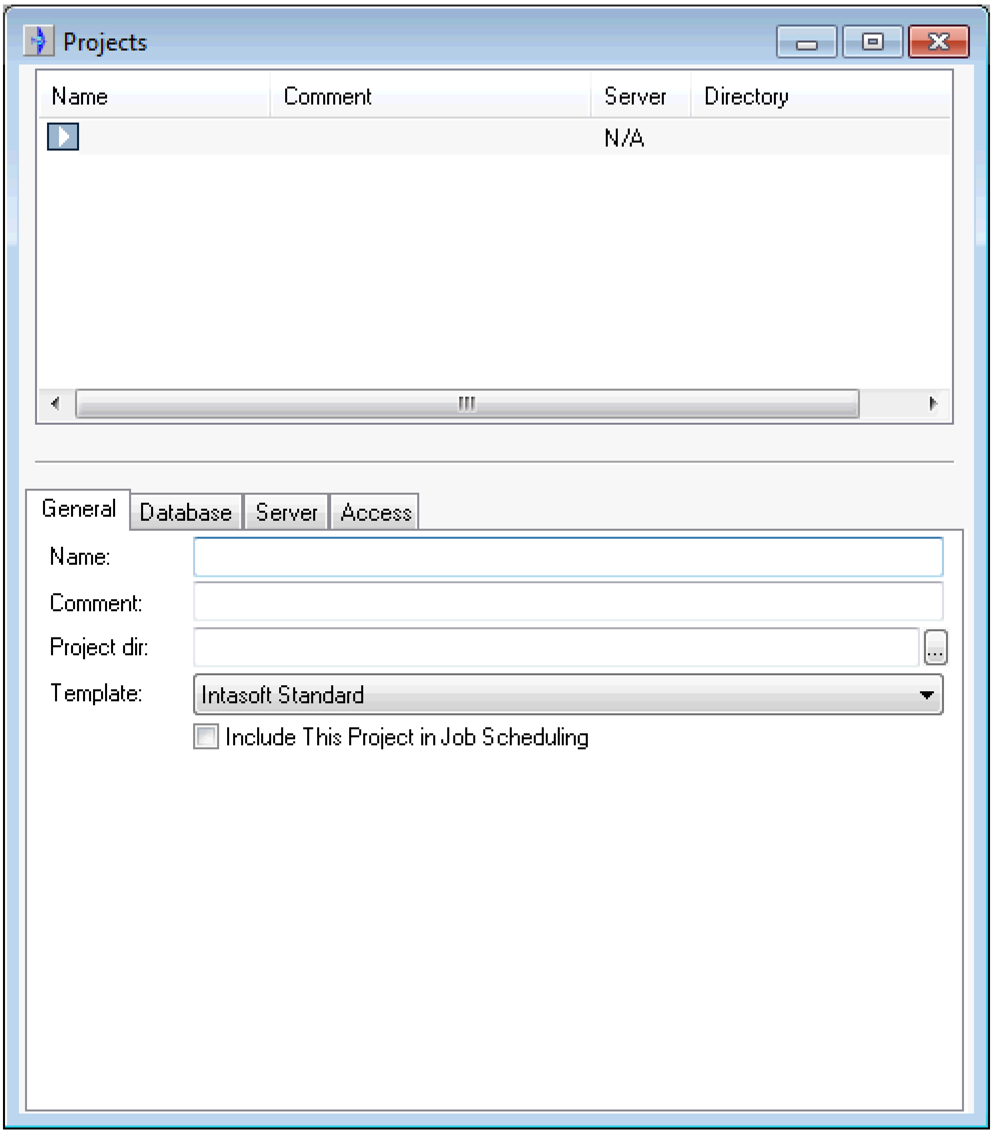
In order to create an ***AllChange*** project you will need to have performed an Administrator Workstation Installation (if you did a full install on this workstation then this will have done this for you).

You will have an ***AllChange*** Quick Start program on your desktop. Using this select the **AllChange Configuration Editor** (or select **Start | All Programs | Intasoft Software | AllChange Configuration Editor**).

This will run the AllChange administration tool (**ACCONFIG**[1] for short).

As you do not have any projects currently defined you will be presented with the Project Definition window.

[1] ACCONFIG, AllChange Configuration Editor , allows the AllChange Administrator to configure most aspects of AllChange



Name	Comment	Server	Directory
		N/A	

Name:

Comment:

Project dir:

Template: Intasoft Standard

□ Include This Project in Job Scheduling

At a minimum information must be entered on the **General** and **Database** tabs.

On the **General** tab:

- Enter a **Name** for the project (e.g. Test or Play), you may also optionally enter a **Comment**.
- The ***Project dir*** should be specified as the location on your filing system where you wish your AllChange project configuration and non database data to be stored, this is known as the **project directory**[1]. If you have installed locally on the workstation then you will probably want your ***AllChange*** project locally as well (e.g. `C:\ACProjects\Test`*)*. If you installed on a server then you will want your ***AllChange*** projects to be (ideally) on the same server as ***AllChange*** is installed but in a subdirectory for the AllChange projects (e.g. `\\myserver\AllChange\AllChangeProjects\Test`). If this directory does not exist, it will be created automatically. If it does exist then it must be an empty directory or **Acconfig** will think that you are connecting to an existing project rather than creating a new one.

  *It is not recommend putting **AllChange** projects as a subfolder of the **AllChange** installation directory (known as the AllChange **system directory**[2]).*

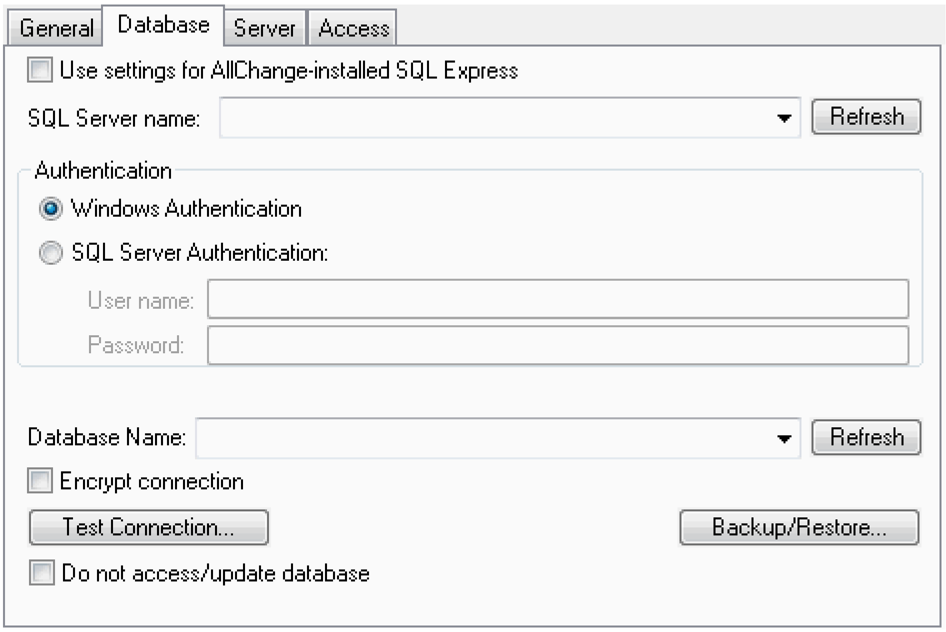
[1] The directory containing the AllChange configuration files and project data

[2] The directory in which AllChange has been installed

- Three templates are supplied for the initial project configuration:
  1. Intasoft Standard - this provides templates for different change management for Documents, Software, Engineering and ITAsset changes
  2. ITIL - this provides ITIL based change and configuration management processes
  3. ICIntegrate - this is intended for use with Intasoft's IntaChange product.

  You should select whichever of these you wish to start with.

On the **Database** tab:



□ Use settings for AllChange-installed SQL Express

SQL Server name:

Authentication

● Windows Authentication

○ SQL Server Authentication:

User name:

Password:

Database Name:

□ Encrypt connection

□ Do not access/update database

- You must specify your **SQL Server name** and **User name** and **Password** if using **SQL server Authentication**. If you selected for ***AllChange*** to install SQL Express when you installed AllChange then you can select **Use Settings for AllChange installed SQL Express** and it will complete these values for you.

  You must also specify a Database name. By convention, ***AllChange*** databases are named **AC_** followed by the project name (with any spaces or punctuation characters replaced by underscores) and this will be entered for you.

- You can use the **Test Connection** button to test the connection to SQL server using the information entered.

When you have entered all the necessary information select **File | Save Window** or the **Save Window** toolbar button (floppy disc picture). A new project with the name given will be created, the directory specified will be created if it does not exist and an empty database created. The project will then be opened in ACCONFIG.

In order to see the projects you have defined or create more projects select **File | Update Projects** in ACCONFIG.

The creation of the first project in this manner creates a basic project definition, if you wish the project directory to be accessed client/server (and you have set up for client/server) then additional information needs to be entered in the project definition.

See AllChange Administrators Manual, Creating and Updating Projects for full details on project definitions.

Having created your project, you now need to set it up for use.

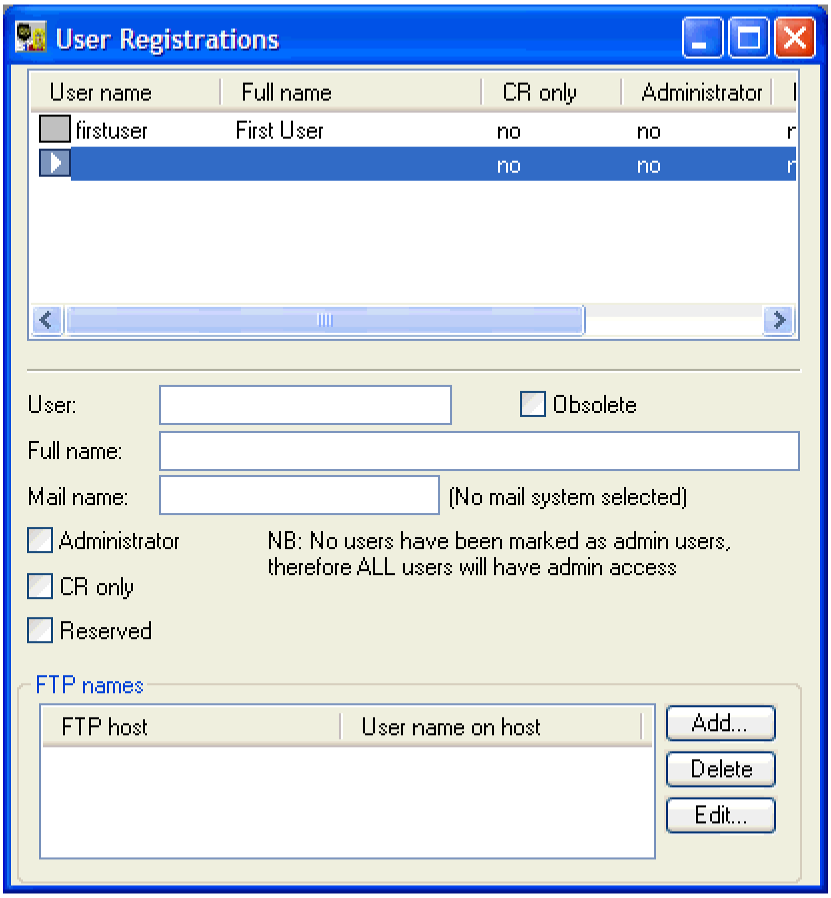
Setting up your project for use

Before an ***AllChange*** project can be effectively used by anyone you need to first register any users who are to use ***AllChange***, and then you need to allocate those users some roles to give them permissions to perform actions in ***AllChange***.

Registering Users

Before anyone can run an ***AllChange*** program they must be a registered user. On installing ***AllChange*** you registered the user performing the install and possibly other users, however this would only register basic information about the user (i.e. their logon name).

In order to view/add/modify user registrations use **Access | User Registration** in ACCONFIG.



User name	Full name	CR only	Administrator	I
firstuser	First User	no	no	r
		no	no	r

User:

□ Obsolete

Full name:

Mail name: (No mail system selected)

□ Administrator

NB: No users have been marked as admin users, therefore ALL users will have admin access

□ CR only

□ Reserved

FTP names

FTP host	User name on host

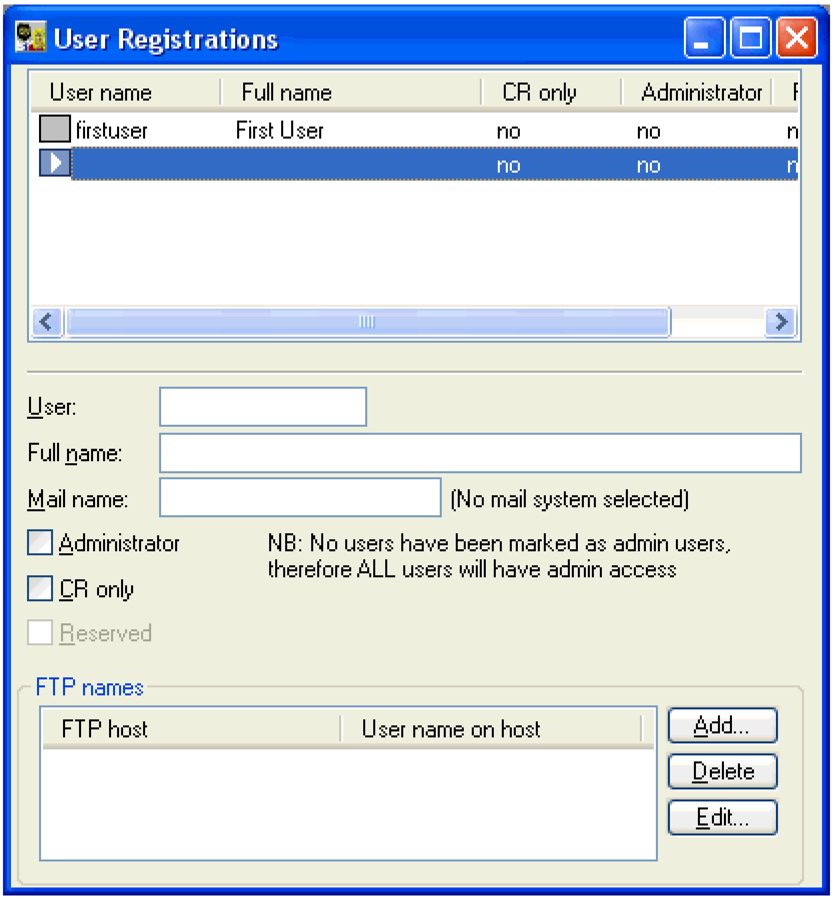
Here you can enter the users Full Name, their email address and other information. See the Administrator Manual, User Registration for full details. You can also use **F1** at any time for context sensitive help.

To modify an entry simply select the user from the list and edit the details in the pane below the list. You should enter the Full Name for yourself and other users registered at installation time and also add their email addresses.

Note that here you select which users are to be classified as Administrators. These should be only users who are to be able to modify the configuration of a project (i.e. users of ACCONFIG). *If no users are classified as administrators then all users may be administrators.*

Note also that whilst all other user registration information is global for all AllChange Projects, Administrative users are per project.

To add a new user select the blank entry at the end of the list and enter the appropriate details.



User name	Full name	CR only	Administrator	F
firstuser	First User	no	no	n
		no	no	n

User:

Full name:

Mail name: (No mail system selected)

☐ Administrator

☐ CR only

☐ Reserved

NB: No users have been marked as admin users, therefore ALL users will have admin access

FTP names

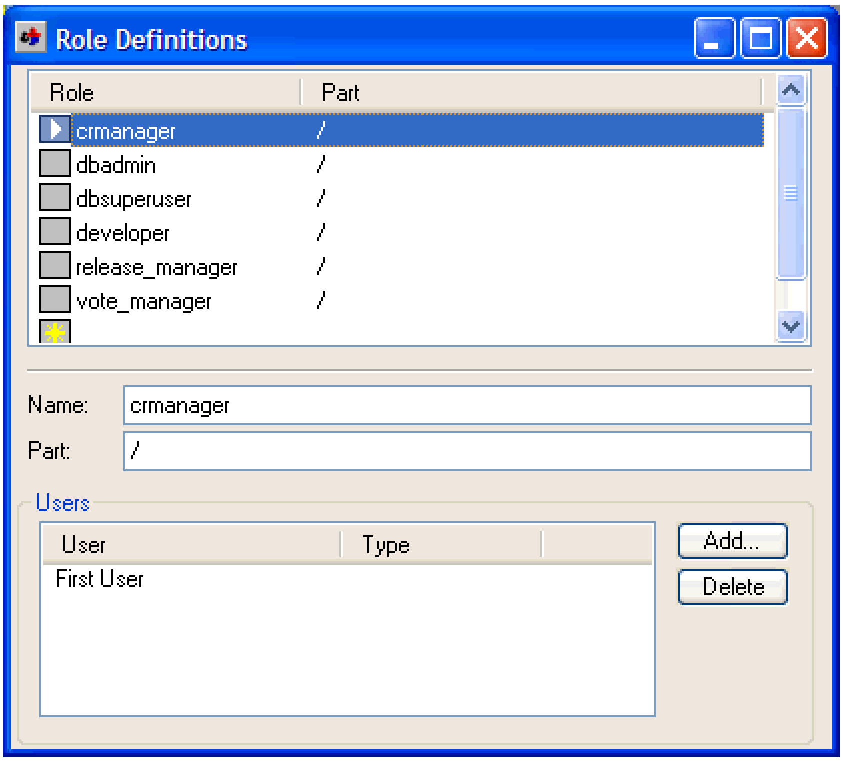
FTP host	User name on host

Don't forget to save your changes, use the floppy disc icon on the toolbar or **File | Save Window**!

Allocating Roles

Having registered your users, you can then allocate roles to them. Roles in AllChange are used to determine who can do what, i.e. each action is permitted by certain roles and users are allocated to those roles to determine what actions each user can perform.

Use **Access | Role Definitions** in ACCONFIG.



Role	Part
crmanager	/
dbadmin	/
dbsuperuser	/
developer	/
release_manager	/
vote_manager	/

Name: crmanager

Part: /

User	Type
First User	

This will show all the roles which are defined out-of-the-box for the project template.

Initially these are defined for all configuration items (**part**[1]s in ***AllChange*** terminology), this may be modified at a later date when you have decided on your parts structure. Similarly, these roles do not have to be used and you may create your own role definitions at a later date.

Initially if you allocate the out-of-the-box roles you can then proceed to use ***AllChange*** and establish what access control rules you wish to put in place.

Select each role in turn and allocate users to it (using the **Add** button next to the Users list which will be initially empty). The special roles dbsuperuser and dbadmin should only be allocated to trusted users, perhaps just initially yourself.

For full details on Roles, simply use F1 when the Role Definitions window is open which will take you to Role Definitions in the AllChange Administrator manual.

Don't forget to save your changes!

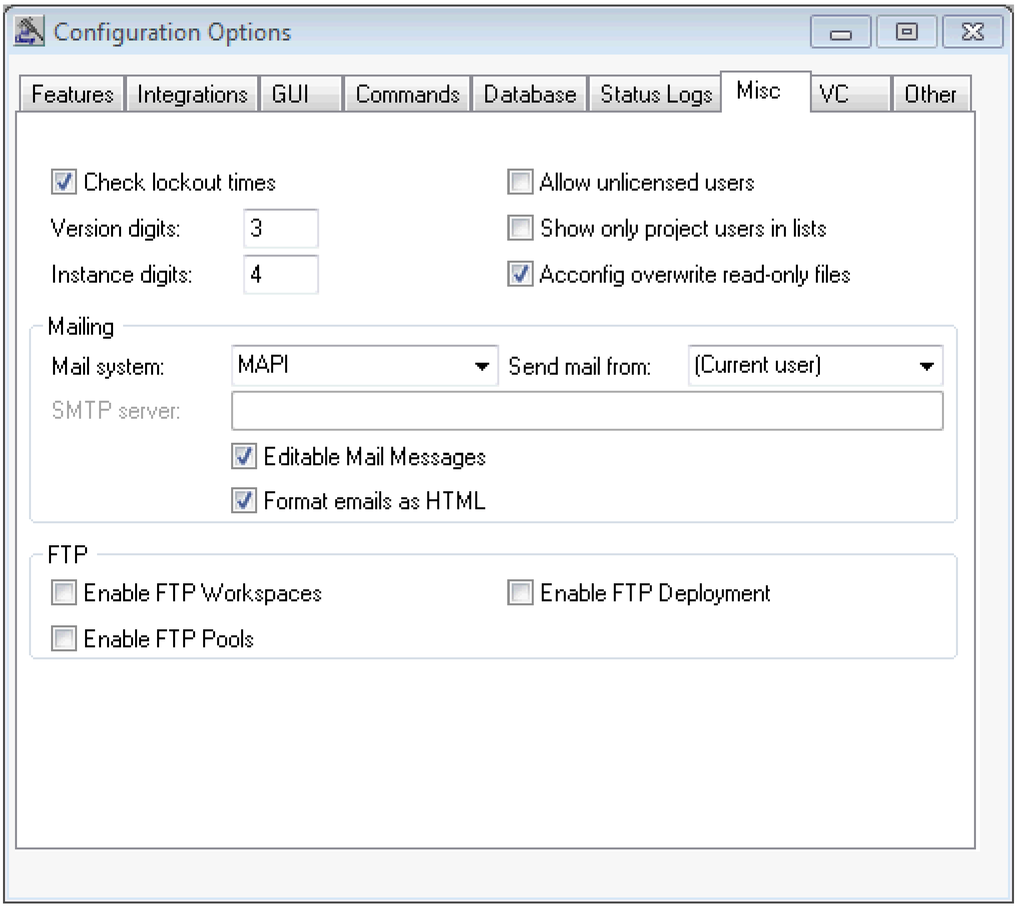
Having registered users and allocated them to roles you can now start to use ***AllChange***, however there are a few more options which you might like to set up first.

Setting up for Email

AllChange has the ability to send emails to users at various stages, for example when a change request is assigned to them. In order for this to take effect emailing must be enabled.

To enable emailing in ACCONFIG, select **Plan | Configuration Options**, then select the **Misc** tab. Select your mailing system from the drop down list.

[1] Parts are all the configuration items known to AllChange. Parts may be of type component or subsystem (or uses a special case). Note that this term may be mapped to another nomenclature (e.g. CI (Configuration Item))



☑ Check lockout times

☐ Allow unlicensed users

Version digits: 3

☐ Show only project users in lists

Instance digits: 4

☑ Acconfig overwrite read-only files

Mailing

Mail system: MAPI

Send mail from: (Current user)

SMTP server:

☑ Editable Mail Messages

☑ Format emails as HTML

FTP

☐ Enable FTP Workspaces

☐ Enable FTP Deployment

☐ Enable FTP Pools

If you use Outlook/Exchange then select MAPI, if you use Lotus Notes, select LotusMail, otherwise select SMTP Mail for generic SMTP Mailing.

Don't forget to save your changes!

AllChange is a very configurable tool

You are now set up and ready to use ***AllChange*** for your test project.

Setting up your Configuration Item Structure

Having set up your ***AllChange*** project using the configuration tool **Acconfig**, we can now run the ***AllChange*** user interface known as ACE. This is available as the **AllChange** button on the QuickStart (or select **Start | All Programs | Intasoft Software | AllChange**).

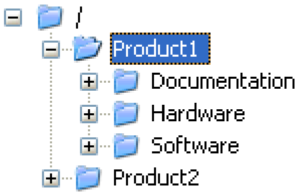
Your configuration items stored in ***AllChange*** are referred to as **part**[1]s and may consist of documentation,hardware or any software object. This is the generic term that ***AllChange*** uses to refer to configuration items although this may be configured as suits the organisation (see Nomenclature Mapping), for example, the ITIL configuration uses the term CI, however the documentation will always use the term part.

There are 2 key types of part

- **subsystem**[2] - this is a folder level part similar to a directory in the filing system
- **component**[3] - this is a document level part similar to a file in a filing system

There may be directories and files associated with your parts (for the files you wish to be put under version control) and there may be no file (for parts such as hardware, IT Assets or manufacturing components that you wish tracked in ***AllChange***).

Initially there is only 1 subsystem present in your database, this is /, the root part (or top of the tree). You need to decide how you wish your parts to be organised. For example you may wish to organise by product, and within that by hardware, software and documentation:



Alternatively you may wish to organise by internal project, or by type (e.g. IT Assets, COTS, In House Software). Any organisation is supported by ***AllChange*** according to your preference.

The organisation will have an impact on the directories in which you work, as these generally reflect the parts hierarchy, but usually from a lower level in the tree. So, for example, you might have your documentation for Product1 in one location on your network and the Software source code in quite another (even on a different server). This is fine and ***AllChange*** allows for this as will be seen later.

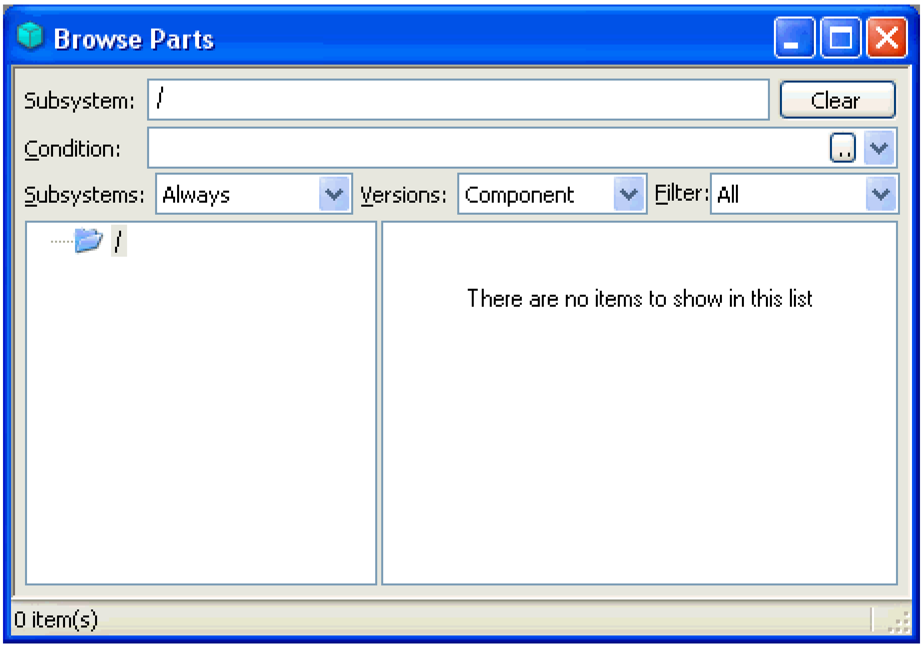
In order to create your initial top level subsystems, you will need developer permission for the top level of the tree. This is discussed in Setting up your First Project.

[1] Parts are all the configuration items known to AllChange. Parts may be of type component or subsystem (or uses a special case). Note that this term may be mapped to another nomenclature (e.g. CI (Configuration Item))

[2] Subsystem-type parts may have children. Subsystem type is analogous to a directory in an operating system filing system. Note that this term may be mapped to another nomenclature (e.g. Folder)

[3] Component-type parts are at the bottom of the part hierarchy and have no children, but they may have versions. Component type is analogous to a file in an operating system filing system. Note that this term may be mapped to another nomenclature (e.g. Item)

To see the parts that you have in your AllChange Project, click on the Parts shortcut or select **Part | Browse**, this will open the part browser window:



Subsystem: /

Condition:

Subsystems: Always Versions: Component Filter: All

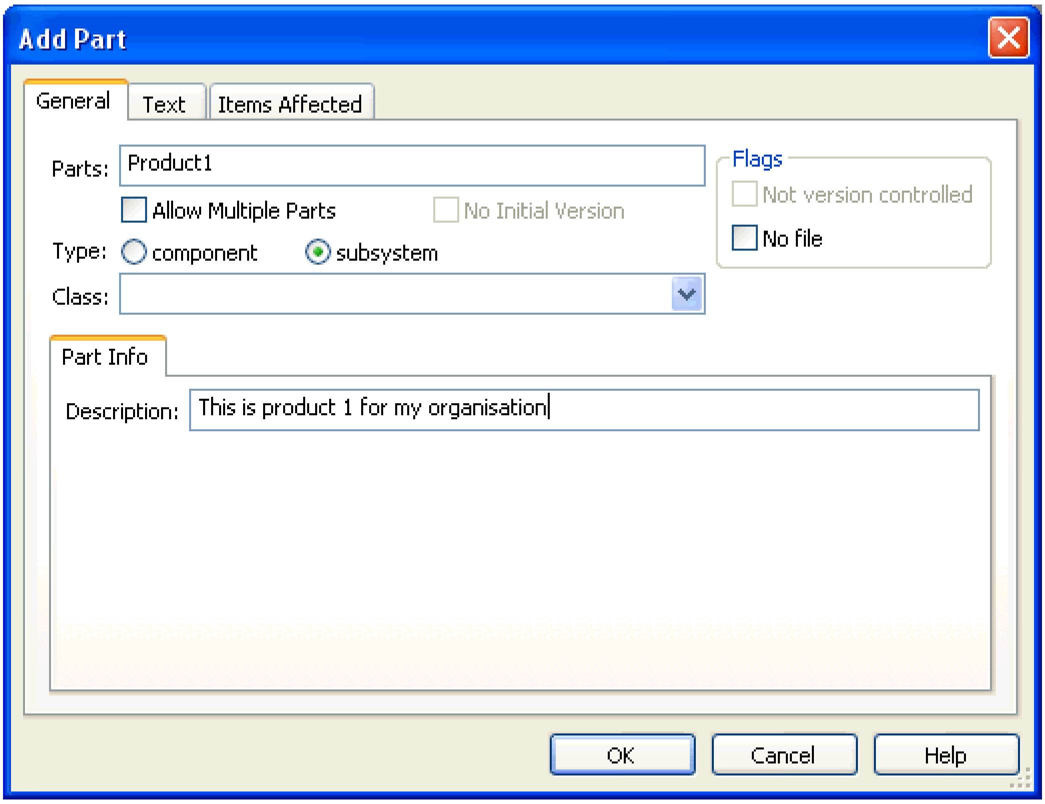
/

There are no items to show in this list

0 item(s)

This is very much like Windows Explorer which allows you to browse your filing system but with the addition of some special filters - for details of the part browser see Browsing the Parts Hierarchy.

To add new subsystems (folders), simply navigate to where you wish to add them as shown in the **Subsystem** control. Then select **Part | Add**, this will open the **Add Part** dialog.



Parts: Product1

□ Allow Multiple Parts □ No Initial Version

Type: ○ component ● subsystem

Class:

Flags
□ Not version controlled
□ No file

Description: This is product 1 for my organisation

Simply enter the name of the folder you wish to create in the **Parts** control, select the **subsystem** as the **Type** and if you wish enter a **Description**, then click OK. The subsystem will be created inside the current Subsystem as shown in the Part Browser.

You can repeat this as many times as you like to create your top level parts structure. When you get to the level in the parts tree below which files are to be stored, then these can be imported automatically from the filing system as will be seen later.

For full details on the **Add Part** dialog see Creating New Parts Explicitly.

Creating your first Change Request

AllChange has a concept of Change Requests (CRs) which are used for any kind of change object that your organisation wishes to use for Change Management. These might be Change Requests, Issues, Problem Reports, Requirements etc. These can be used to represent any kind of form you have for managing change and can be configured to your specific requirements.

The term Change Request (CR) is a generic term used by ***AllChange*** for all these types of forms/change objects and may be may be configured as suits the organisation (see Nomenclature Mapping), for example, the ITIL configuration uses the term ACMD (AllChange Management Document), however the documentation will always use the term CR.

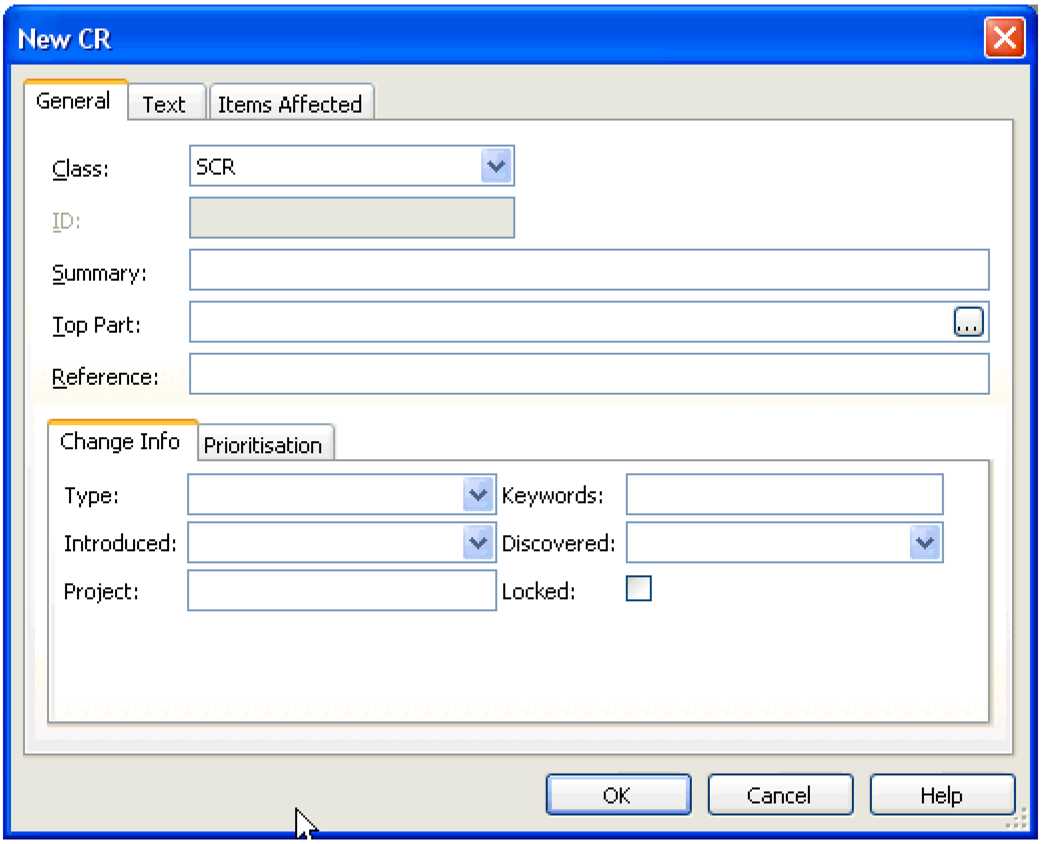
Furthermore, CRs may be of different types, known as the **class**[1], for different change management objects (for example the Intasoft Standard configuration has SCR (Software Change Request) and DCR (Document Change Request) among others and the ITIL configuration has Incident, Problem and RFC), each of which has a different set of data fields and process associated with it. Classes are entirely definable as required by your organisation and some example classes are defined for the project template that was used to create your ***AllChange*** project, see Classes.

You can use CRs to authorise any change within your organisation, whether change to the organisation such a new employee, a change to IT infrastructure, change to software developed in house or change to documents etc. Any type of change can be tracked and authorised using CRs.

Change can also include change to an existing object (part), or the addition of new objects (parts). You can link the parts changed/added to the CR authorising them and this link can be created automatically for you on putting the new part (new version of an existing one) into ***AllChange*** (on check in). So if we wish to add or change any parts which are to be controlled by CRs a CR must exist first and must be suitably authorised.

To create a CR select **CR | New** from the menu:

[1] Classes are used to classify parts, change requests and baselines and to determine their life-cycle. They are identified by a site-supplied name. This can be used to classify the item, e.g. source code, documentation or hardware for a part, bug fix or upgrade request for a change request, test or release for a baseline. Note that this term may be mapped to another nomenclature (e.g. CIType)



Class: SCR

ID:

Summary:

Top Part:

Reference:

Type:

Keywords:

Introduced:

Discovered:

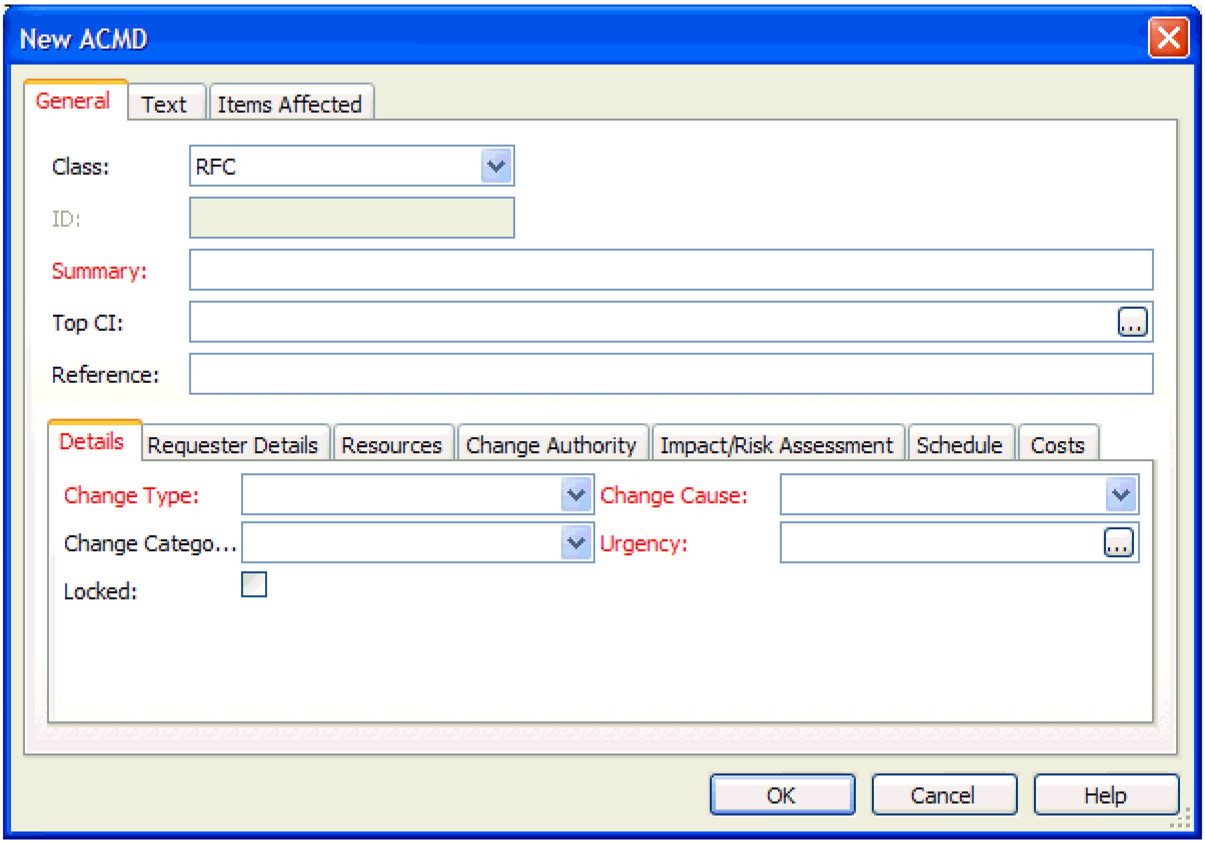
Project:

Locked: □

The first thing to select is the **Class**[1] of CR you wish to create. Those initially available will be those defined in the out-of-the-box configuration that you selected when you created your AllChange project. The example above shows an SCR (Software Change Request) from the Intasoft Standard configuration.

Below is an example of an RFC from the ITIL configuration:

[1] Classes are used to classify parts, change requests and baselines and to determine their life-cycle. They are identified by a site-supplied name. This can be used to classify the item, e.g. source code, documentation or hardware for a part, bug fix or upgrade request for a change request, test or release for a baseline. Note that this term may be mapped to another nomenclature (e.g. CIType)



Class: RFC

Any fields shown in red are mandatory and must have a value before the CR can be created.

Each class has certain fields in common, these are:

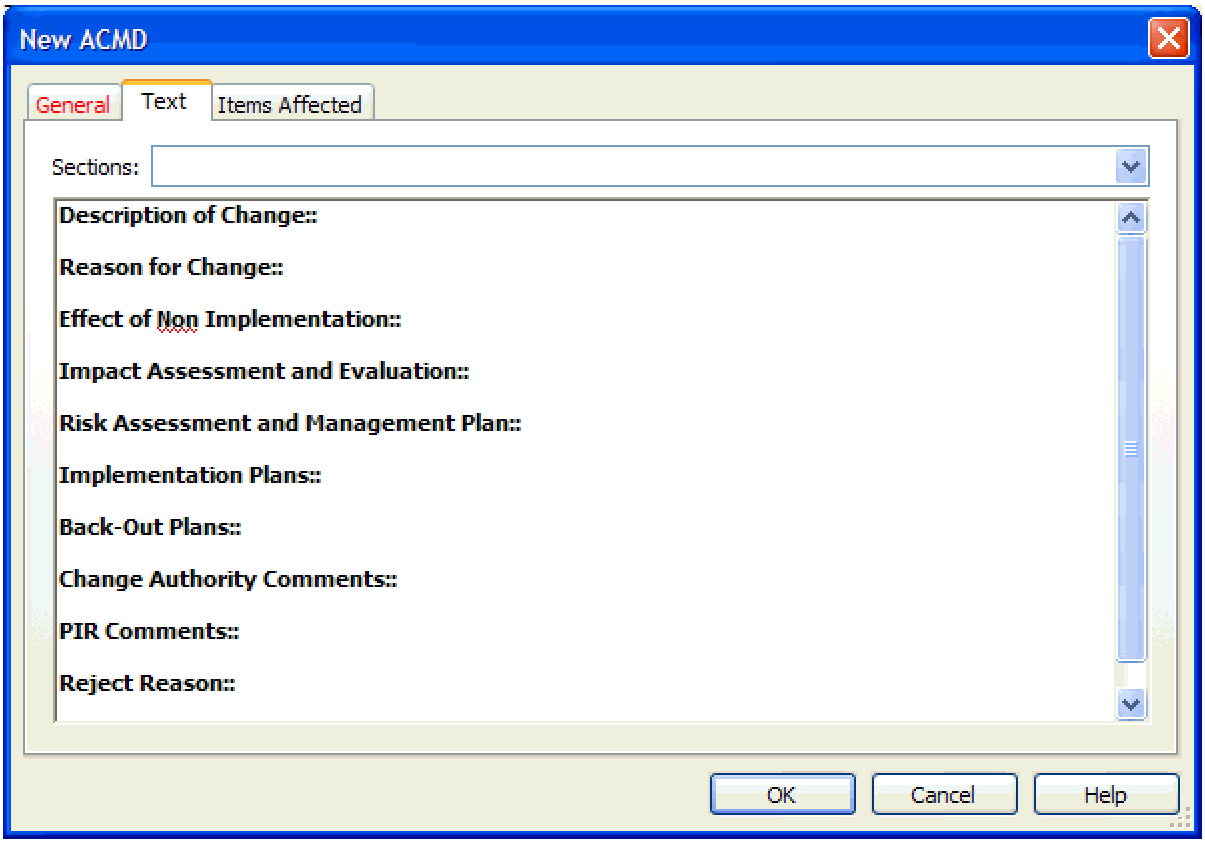
- **ID** - This is only used if the CR requires a user entered element to be allocated to the CR number/identifier. If this is not required for the class it will be disabled.
- **Summary** - This should be used to summarise the required change. This field may be renamed for your organisation (for example, Title) if this is required, see Field Name Definitions.
- **Top Part** - This may be used to indicate the highest point in the parts tree that this CR affects. For example, it might be used to show that this change affects only Product1 and only the Software in Product1. This field may also be renamed if this is required. You will note that for the ITIL configuration this is shown as **Top CI** since in ITIL parts have been renamed to CIs.
- **Ref** - This is a simple text field typically used to store a cross-reference to another system; for example a help desk identifier. This field may also be renamed if this is required.

The other fields shown on the General tab are totally customisable per class of CR, these are known as arbitrary fields. As can be seen above the fields are different in the Intasoft Standard SCR and the ITIL RFC. These fields may be changed according to your organisations requirements for change management documents, see Field Name Definitions.

Any fields shown in red are mandatory on creation of the CR and a value must be entered/selected. Other fields may be completed now or at a later date.

For details of these (arbitrary) fields for the Intasoft Standard configuration see CR Fields for Standard Project Template, for the ITIL configuration see ACMD Fields (ITIL).

The **Text** tab of the **New CR** dialog allows you to add larger quantities of text associated with the CR such as the description, in a predefined template format divided into sections. The template format used is entirely definable as required for your organisation, see About Classes.



Description of Change::

Reason for Change::

Effect of Non Implementation::

Impact Assessment and Evaluation::

Risk Assessment and Management Plan::

Implementation Plans::

Back-Out Plans::

Change Authority Comments::

PIR Comments::

Reject Reason::

The **Items Affected** tab allows you to associate other items with the CR including File **Attachments** and links to other elements of the ***AllChange*** database such as parts and baselines and even links to other CRs. These can be left blank in the out of the box configurations when the CR is created.

On selecting **Ok** a new CR will be created with the data specified in the dialog. An identifier will be automatically allocated to the CR although what this is composed of is, as you would expect by now, customisable for your organisation according to the class, see About Classes.

In order to see the CR's you have in your AllChange project database, select the CRs icon on the shortcut bar or select **CR | Browse**, this will open the CR Browser window which allows you browse and query the CRs you have. Various filters are available to allow selective browsing, see Browsing CRs for details.

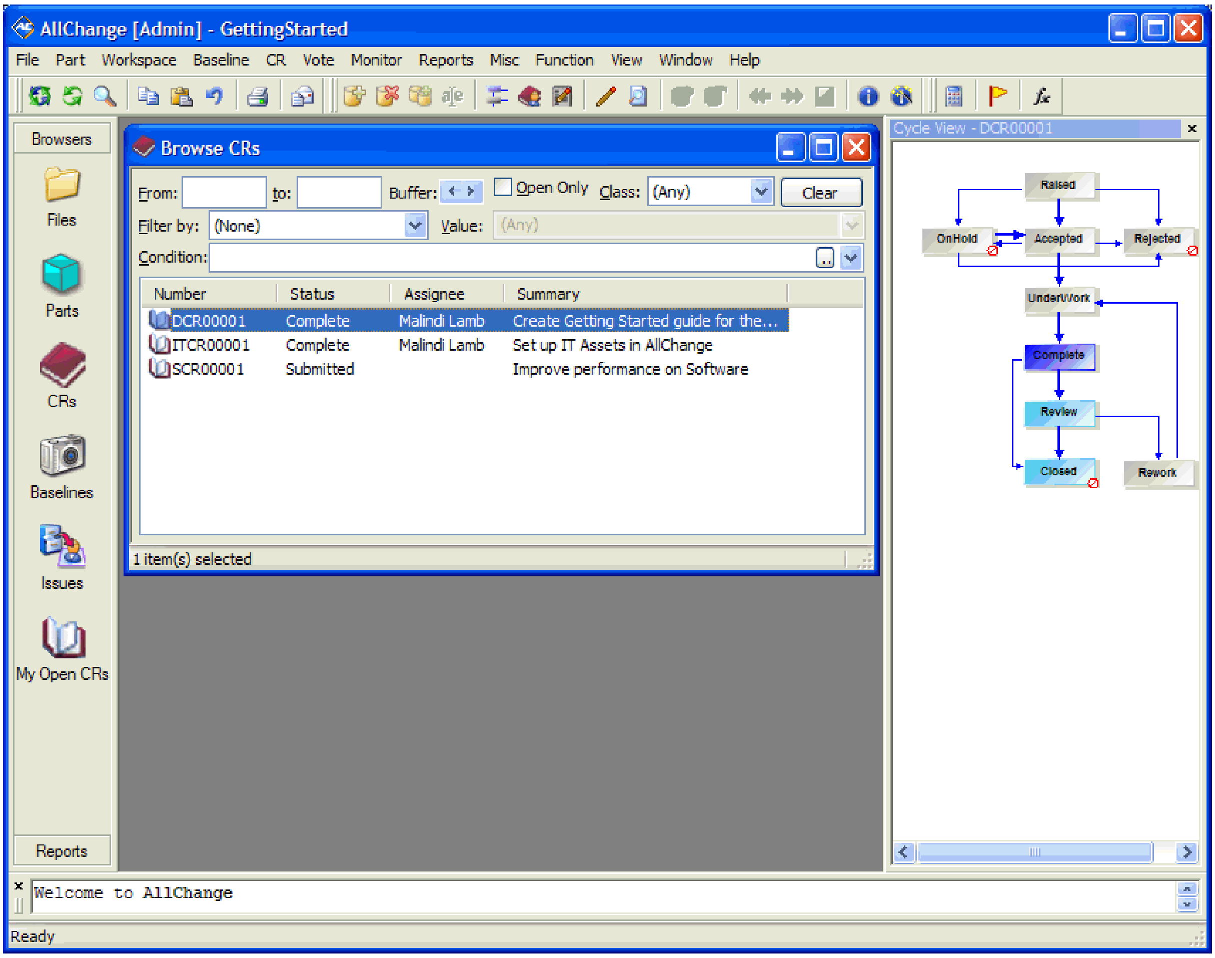
To view the details of a specific CR, simply double click on the selected item in the CR browser, or select **CR | View** and the **CR Viewer** window will open for the selected CR showing all information associated with the CR on various tabs.

See Change Requests for more information.

Change Requests

Creating your first Change Request introduced the concept of Change Management in ***AllChange***. Here we will look in more detail at CRs.

To browse the CRs in your ***AllChange*** project (remember these might be referred to by another name (e.g. ACMD in the ITIL configuration), simply select the CRs button on the shortcut bar or folder list, or select **CR | Browse**).



From: to: Buffer: □ Open Only Class: (Any)

Filter by: (None) Value: (Any)

Condition:

Number	Status	Assignee	Summary
DCR00001	Complete	Malindi Lamb	Create Getting Started guide for the…
ITCR00001	Complete	Malindi Lamb	Set up IT Assets in AllChange
SCR00001	Submitted		Improve performance on Software

1 item(s) selected

```
Welcome to AllChange
```

Ready

The screen shown above is from a project with the Intasoft Standard configuration. Each CR has associated with it a life-cycle which implements the change management process for that class of CR. The screen shows the cycle for a Document CR (DCR). Each class may have a different process associated with it and it is of course entirely definable by your organisation. The project templates supplied provide a number of cycles (processes) as suggestions for the way in which you may wish to work, containing the basic necessities for a good change management process which you may use as a starting point and tailor to your requirements or you may want to create your own from scratch

The ITIL configuration will look a little different as CRs have been renamed to ACMDs and the class of CR and its life cycle are different.

Each CR will be shown with an icon to indicate whether it is an open CR or a closed CR . Each CR has a **life-cycle**[1] associated with it and each status in the life-cycle is defined as being either open or closed. Thus whether a CR is open or closed depends on its *current* **status**[2] within the life-cycle. Life-cycles are of course entirely definable by your organisation to suit your own procedures and processes, see Life-Cycles. A number of life-cycles are provided with the project templates to provide examples and possibly a starting point from which you can make modifications to tailor them to your organisations requirements.

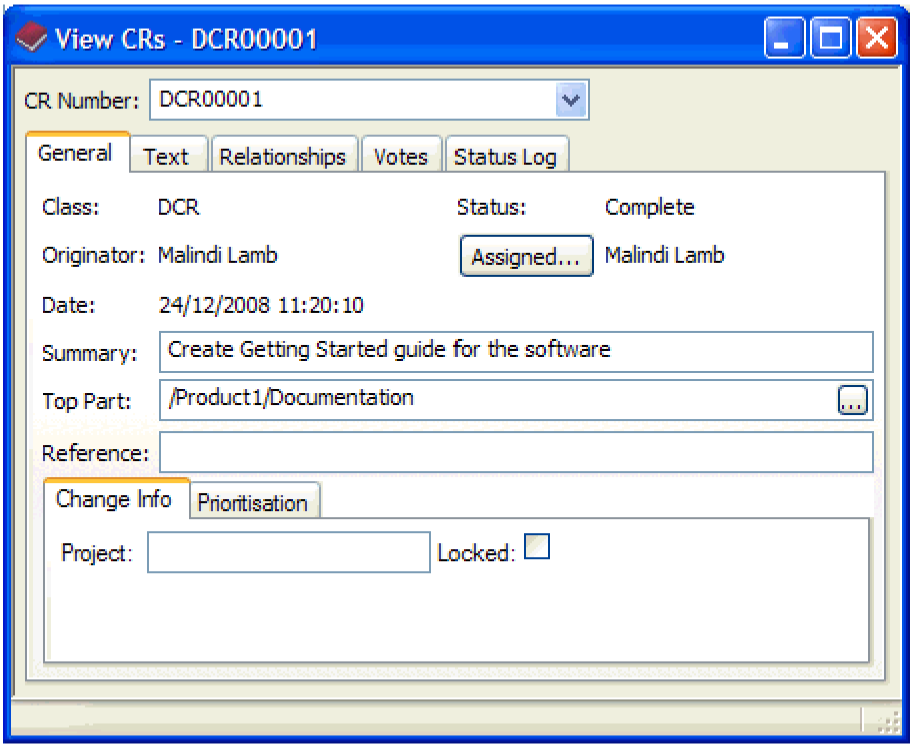
Various filters are available at the top of the browser window to enable you to limit the selection of CRs that you see, for full details see Browsing CRs.

Each CR has a concept of an assignee, the person/group that is currently responsible for the CR. Thus a common query or filter might be to show the CRs currently assigned to the user and which are open.

Various other information about the CR is shown in the browser, by default this will be the current **Status**, the current **Assignee** and the **Summary**. Which columns of information are shown may be tailored to individual preferences, see Defining Browser Content.

In the Cycle View pane of the ***AllChange*** screen the life-cycle for the currently selected CR will be shown, if one (and only one) CR is selected. The colour scheme will (by default) show the current status, any statuses which are valid progressions from the current status and any statuses which are not currently valid progressions. The colour scheme used, the zoom level and various other options may be changed according to user preferences, see Cycle View.

To view the details of a specific CR, simply double click on the selected item in the CR browser, or select **CR | View** and the **CR Viewer** window will open.



CR Number: DCR00001

Class: DCR

Status: Complete

Originator: Malindi Lamb

Assigned... Malindi Lamb

Date: 24/12/2008 11:20:10

Summary: Create Getting Started guide for the software

Top Part: /Product1/Documentation

Reference:

Project:

Locked: □

[1] A life-cycle (or simply cycle) is defined as a series of statuses through which a part, CR or baseline passes and may be used to control access to items and implement approval procedures etc.

[2] A status is a stage in a life-cycle through which parts, change requests or baselines may pass. They may be used to control access to items and implement approval procedures etc. Note that this term may be mapped to another nomenclature (e.g. State)

On the **General** tab this will show:

- **Class** - this defines the **life-cycle**[1], arbitrary fields and Text of the CR. The CR shown in the screen above is of class DCR (Document Change Request) in the Intasoft Standard configuration. Classes are of course entirely definable as required by your organisation, see Classes.
- **Status** - this is the *current* **status**[2] in the **life-cycle**[3] of the CR. In the screen shown above this is the initial status of the life-cycle for a DCR (Document Change Request)- Raised. The status may be changed by progressing it through its life-cycle to any of the valid progressions from this point by double clicking on an appropriate status in the **Cycle View** window. See also CR Life-Cycles.
- **Originator** - this Identifies the originator of the CR. This field is automatically set to the user who created the CR.
- **Assignee** - this is the user/group currently assigned to the CR. This may be set by selecting the **Assigned** button, but may also be configured to be set automatically as the CR progresses through the life-cycle (see Predefined Actions in Life-Cycles in the Administrator Manual). On assignment to another user, by default, an email is sent to the assignee to inform them of this, providing email integration has been enabled.
- **Summary** - This should be used to summarise the required change. This field may be renamed for your organisation (for example, Title) if this is required, see Field Name Definitions.
- **Top Part** - This may be used to indicate the highest point in the parts tree that this CR affects. For example, it might be used to show that this change affects only Product1 and only the Documentation in Product1. This field may also be renamed if this is required. You will note that for the ITIL configuration this is shown as **Top CI** since in ITIL parts have been renamed to CIs.
- **Ref** - This is a simple text field typically used to store a cross-reference to another system; for example a help desk identifier. This field may also be renamed if this is required.

The remainder of the fields are arbitrary fields which vary according to the class of CR. Up to 100 of these may be defined per class and placed on any named tab of your choice, see Field Name Definitions. Those shown in the screen above are those defined for the DCR class of CR in the Intasoft Standard project configuration.

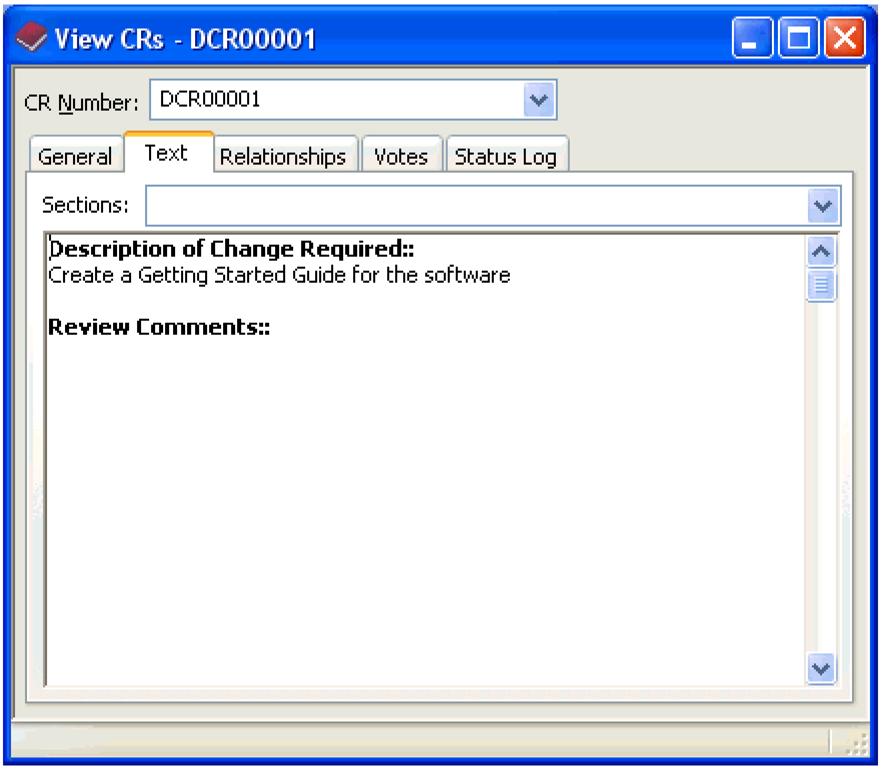
Fields which may be changed using controls on the CR viewer may be modified as required and then select **Misc | Update** or the 💾 toolbar button to save your changes.

On the **Text** tab a single edit control will be shown containing various sections according to the class of the CR. The sections may be defined as required by your organisation and associated with the class, see Classes.

[1] A life-cycle (or simply cycle) is defined as a series of statuses through which a part, CR or baseline passes and may be used to control access to items and implement approval procedures etc.

[2] A status is a stage in a life-cycle through which parts, change requests or baselines may pass. They may be used to control access to items and implement approval procedures etc. Note that this term may be mapped to another nomenclature (e.g. State)

[3] A life-cycle (or simply cycle) is defined as a series of statuses through which a part, CR or baseline passes and may be used to control access to items and implement approval procedures etc.



CR Number: DCR00001

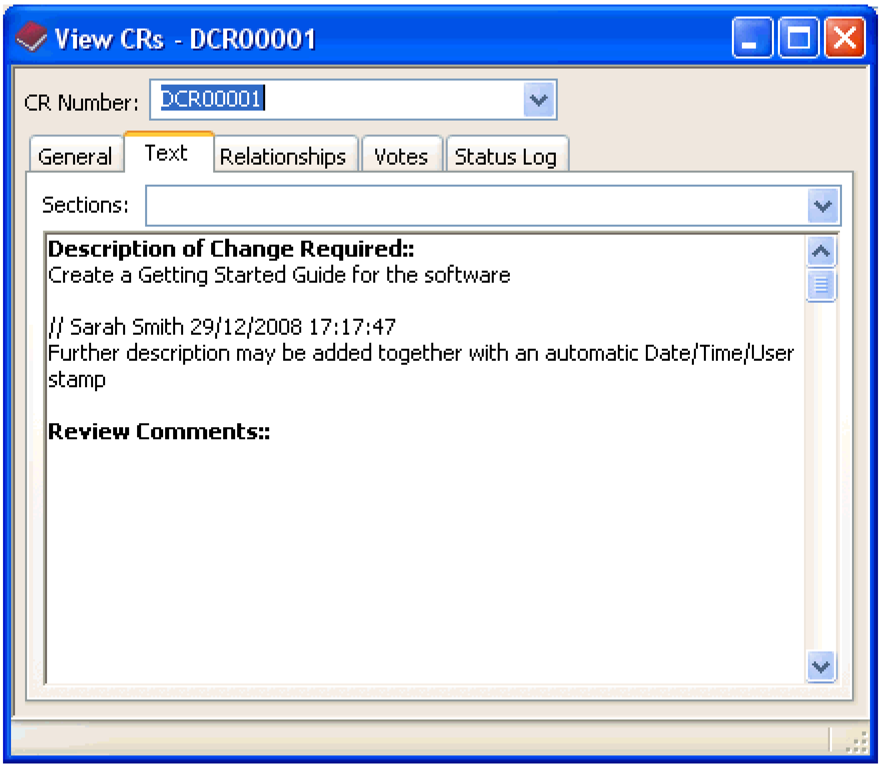
Sections:

Description of Change Required::
Create a Getting Started Guide for the software

Review Comments::

For most classes in the supplied configurations, to modify the text, simply enter the required text and select the 💾 toolbar button to save your changes.

Text may also be appended to the end of a section together with a Date/Time/User stamp by using **Insert Into Text** from the right click context menu or the CR menu.



CR Number: DCR00001

Sections:

Description of Change Required::
Create a Getting Started Guide for the software

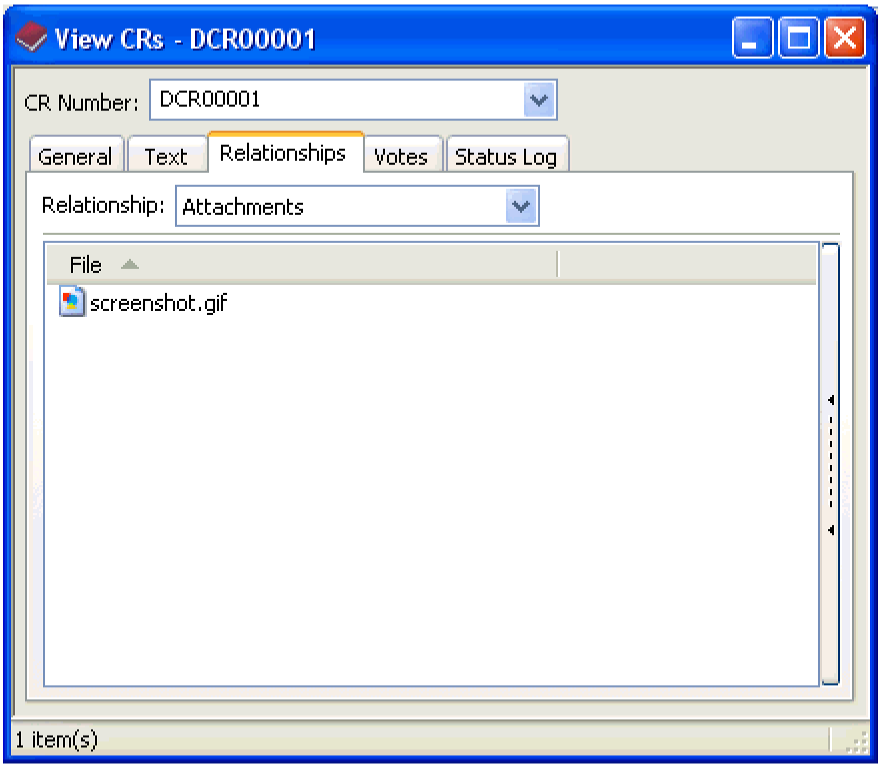
// Sarah Smith 29/12/2008 17:17:47
Further description may be added together with an automatic Date/Time/User stamp

Review Comments::

Certain classes may be marked as having **Protected Text** (e.g. **WorkOrder** in the Intasoft Standard Configuration), this means that text may not be simply entered into the Text control on the Text tab, but may only be appended to the end of a section using **Insert Into Text**, unless you have a certain privileged role (crmanager or dbsuperuser in the Intasoft Standard configuration).

The **Relationships** tab shows relationships to other elements in the ***AllChange*** database including relationships to other CRs and parts and baselines which are affected by the change. It also allows files to be attached to the CR and files which are attached to be viewed.

Select **Attachments** on the Relationships drop down list to view attached files:



CR Number: DCR00001

Relationship: Attachments

screenshot.gif

1 item(s)

To view an attachment simply double click on the file.

To add a new attachment or detach a file, the right click context menu may be used, or the slide out button may be used to reveal **Attach**/**Detach** buttons.

For full details of attachments see The Relationships Tab.

CR Life-Cycles

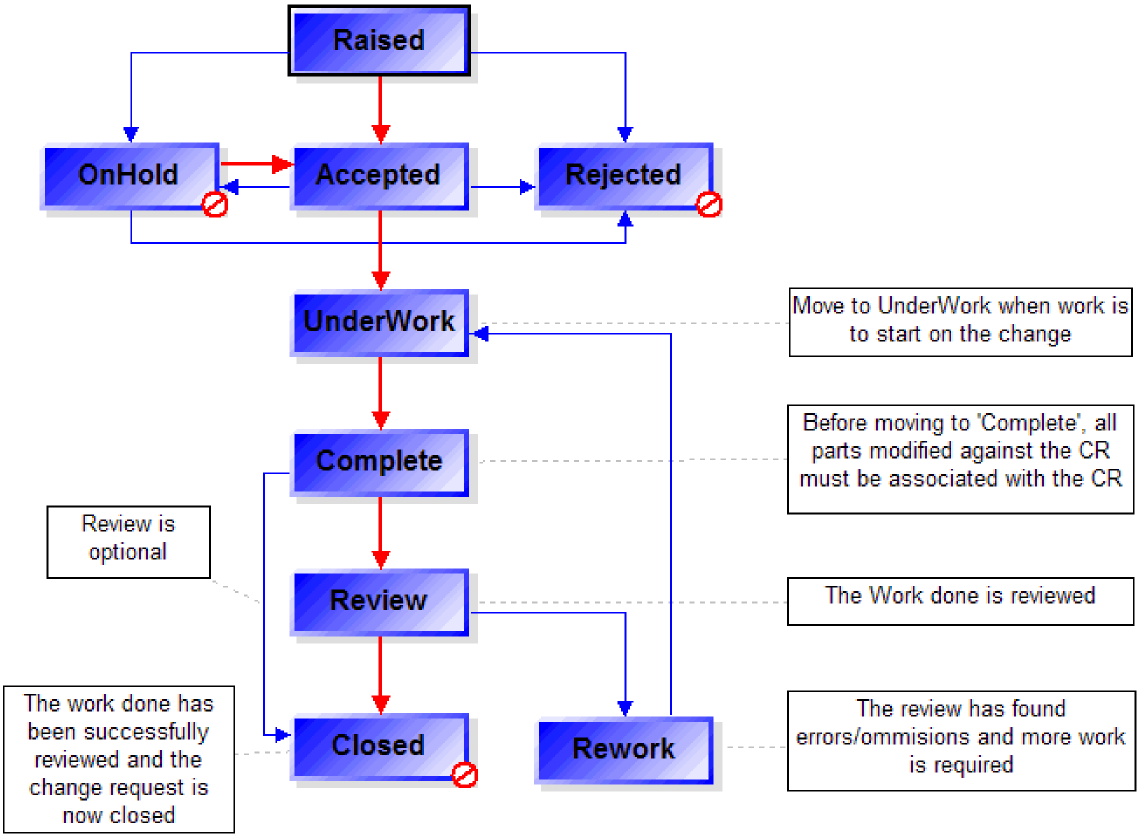
CRs have **life-cycle**[1]s associated with them according to their class. This may be used to implement approval procedures and track the progress of the change. Each CR has a current status within the life-cycle to denote which stage in the process has been reached.

Life-cycles are, as you would expect, entirely definable by your organisation and each class of CR may have a different life-cycle to reflect different change processes. In the supplied project configurations various cycles are provided and associated with the classes defined for that project. These can give you a good starting point for tailoring to your specific requirements. Attributes can be associated with statuses in the life-cycle and conditions for entry into status. These can be used to enforce your change management procedures. Furthermore, actions can be automated on progression into a status (e.g. sending email).

A fairly typical simple change process is implemented in the supplied Intasoft Standard project configuration for the DCR class:

[1] A life-cycle (or simply cycle) is defined as a series of statuses through which a part, CR or baseline passes and may be used to control access to items and implement approval procedures etc.

generic_cr_cycle Change Management process for CRs of class ECR, DCR, ITCR, DoorsRequirement



Move to UnderWork when work is to start on the change

Before moving to 'Complete', all parts modified against the CR must be associated with the CR

Review is optional

The Work done is reviewed

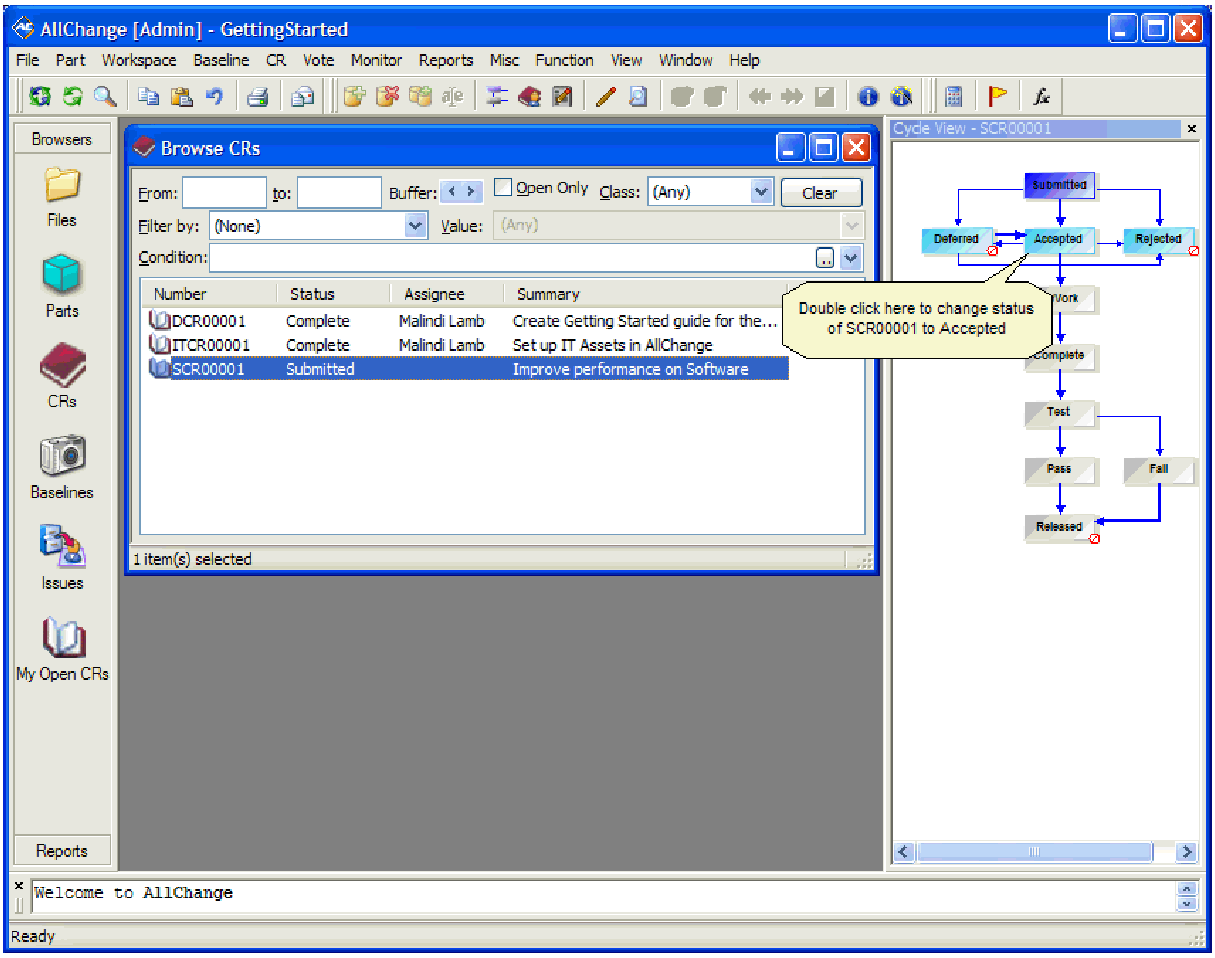
The work done has been successfully reviewed and the change request is now closed

The review has found errors/ommisions and more work is required

Initially the DCR is **Raised**, then it undergoes an approval process which decides whether to accept, reject or defer the change. Once it has been **Accepted**, it must then be progressed to **UnderWork** when work starts on the change. Once the work is finished this should be indicated by progressing the status to **Complete**. It may then go through an optional review process or be closed.

For full details see CR Change Management Process for Intasoft Standard Project Template.

To progress a CR through its life-cycle simply select the CR in the CR Browser or open it in the CR Viewer and double click on the desired status in the cycle viewer:



From: to: Buffer: □ Open Only Class: (Any)

Filter by: (None) Value: (Any)

Condition:

Number	Status	Assignee	Summary
DCR00001	Complete	Malindi Lamb	Create Getting Started guide for the…
ITCR00001	Complete	Malindi Lamb	Set up IT Assets in AllChange
SCR00001	Submitted		Improve performance on Software

Double click here to change status of SCR00001 to Accepted

1 item(s) selected

Welcome to AllChange

Ready

Using a Change Request to Authorise Changes

AllChange has the ability to enforce that an approved CR is associated with any changes to parts/files (of a class that require this).

In order to use a CR for this purpose, the CR must be in an acceptable status as defined in the life-cycle for the CR, and also, by default, must be assigned to the person wishing to change the part/file.

In the Intasoft Standard configuration this is the **UnderWork** status for DCRs, ITCRs and ECRs, it is the **InWork** status for SCRs.

In the ITIL configuration it is the **Approved** status for RFCs.

This means that in order to add or change some files (parts) to ***AllChange*** of a class which requires a CR you will need to progress the status of the CR to the appropriate status and also assign it to yourself.

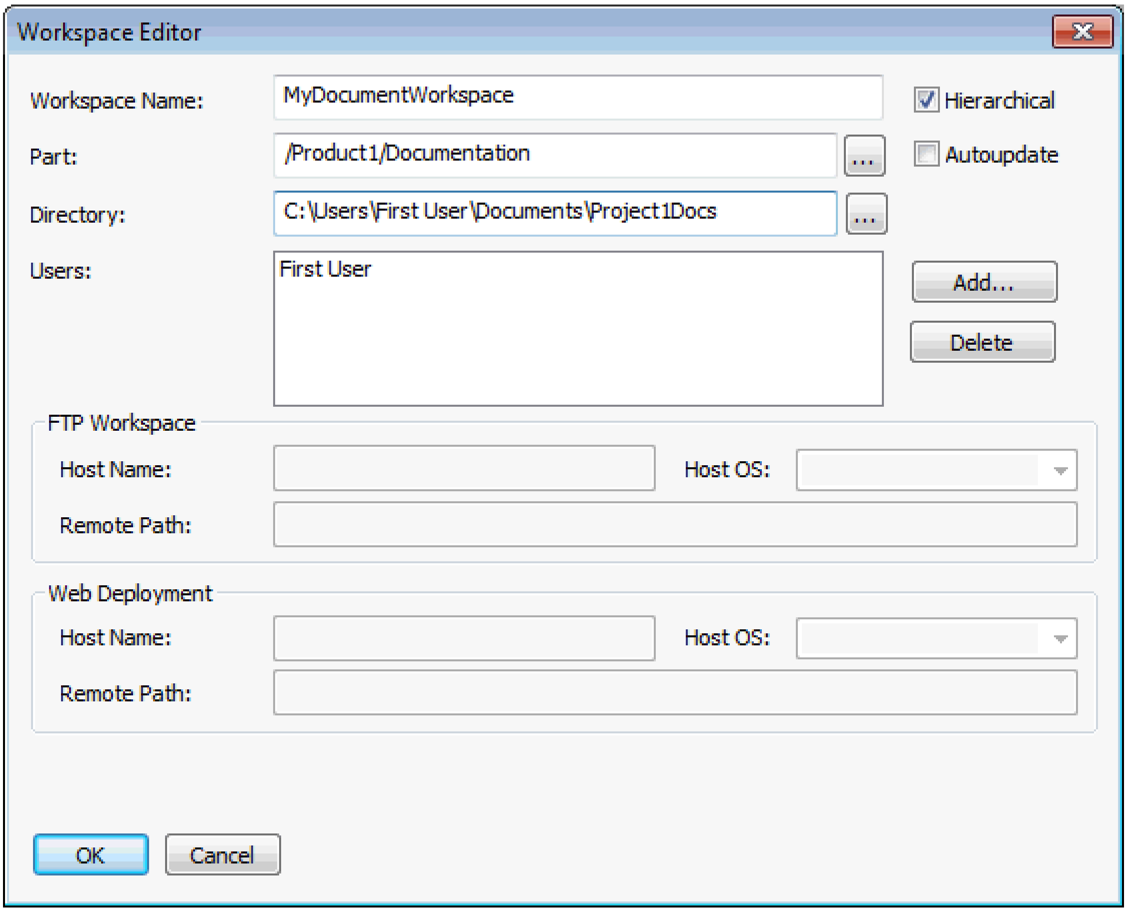
Storing some files under AllChange Control

Setting up your Configuration Item Structure introduced the concept of parts and creating the top level parts structure to reflect how you wish to organise your configuration items.

Remember that the term part may have been redefined (e.g. for the ITIL configuration the term CI is used).

The next step is to put some files that you wish to be under ***AllChange*** control into ***AllChange***. These may be source code, MS Word documents, spreadsheets, pictures or any other kind of file. ***AllChange*** needs to know where these files are and where you would like them put within the parts tree. ***AllChange*** uses the concept of a **workspace**[1] to provide this information.

First we must define the workspace, To do this use **Workspace | Add Workspace**.



Workspace Editor

Workspace Name: MyDocumentWorkspace

☑ Hierarchical

Part: /Product1/Documentation

□ Autoupdate

Directory: C:\Users\First User\Documents\Project1Docs

Users: First User

Each user may have as many workspaces as they wish. It is sensible to consider a naming convention for workspace to ensure that they are easily identifiable for each user (e.g. prefix or suffix each workspace name with the users name or initials). The name of a workspace may not contain spaces.

The **Part** associated with the workspace identifies where in the parts tree files from this workspace will be placed. There may be as much of a directory structure as required beneath this part, this just indicates the highest level.

[1] Workspaces are used to hold local copies of parts for examination, editing and building purposes. The objects held in a workspace are operating system files and the workspaces themselves are simply operating system directories. Note that this term may be mapped to another nomenclature (e.g. Sandpit)

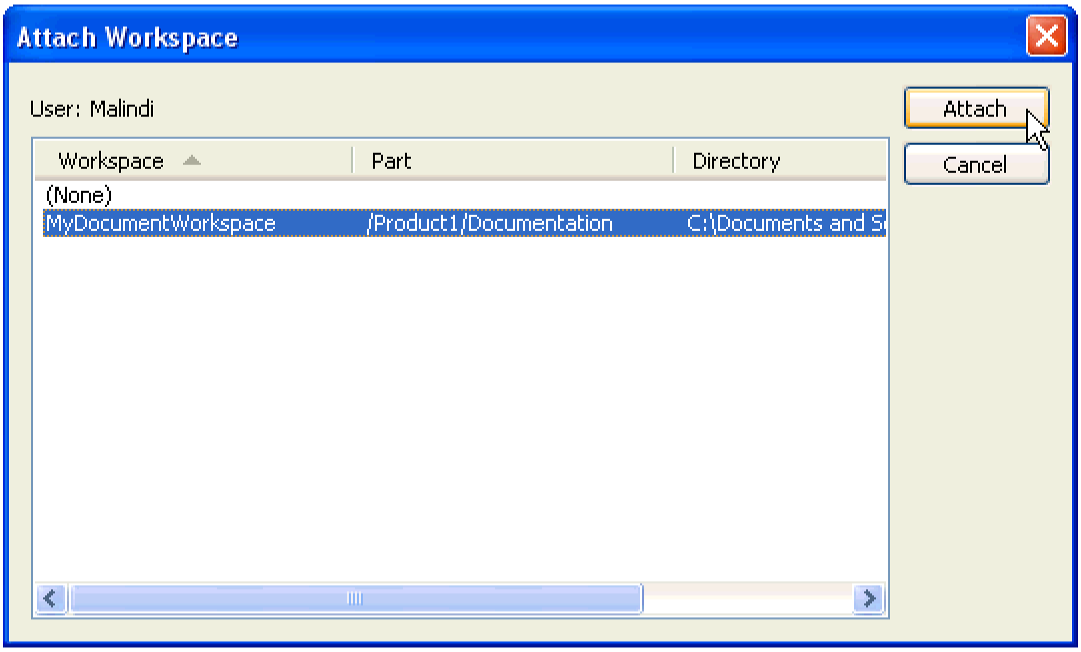
The Directory associated with the workspace identifies the location on the filing system for this workspace. Files and Directories below this will be associated with parts below the workspace **Part** specified above. The directory tree will (by default) be mirrored in the parts tree. This should be the directory containing the files you wish to place under ***AllChange*** control, it may be anywhere on your local hard drive or anywhere on the network.

Many different workspaces may be associated with the same part allowing copies of parts within the tree to be placed in multiple locations on the filing system (e.g. for different users, different environments - development, test, production).

The example in the screenshot above creates a workspace for /Product1/Documentation in My Documents\Product1Docs (C:\Users\<user name>\Documents\Product1Docs in Windows 7) directory.

For further details on workspace see About Workspaces.

Having defined our workspace we now need to tell ***AllChange*** that this is the workspace we wish to use, we need to *attach* to the workspace. Simply select **Workspace | Attach Workspace**, select the workspace you wish to attach to and click the **Attach** button.

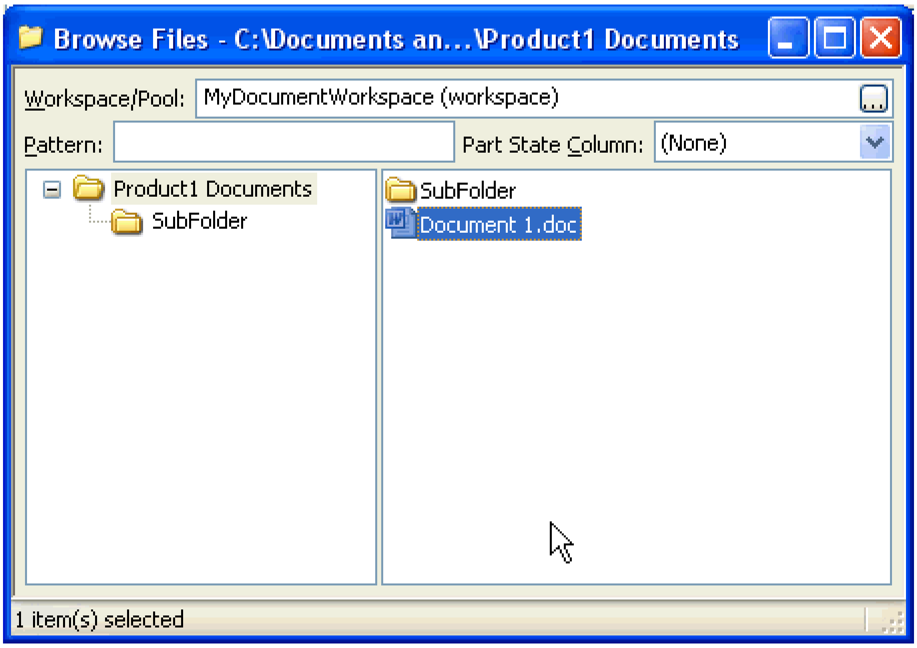


User: Malindi

Workspace	Part	Directory
(None)		
MyDocumentWorkspace	/Product1/Documentation	C:\Documents and S

The workspace you are currently attached to is shown in the bottom right of the ***AllChange*** window.

Now, to put files which are in your workspace directory under ***AllChange*** control we need to *check them in*.There are several ways to do this but one of the simplest is to open the File Browser using the **Files** shortcut or **File | Browse**.This will open a window allowing you to browse the filing system, by default it will show you the files and directories in the workspace you are currently attached to.



Browse Files - C:\Documents an...\Product1 Documents

Workspace/Pool: MyDocumentWorkspace (workspace)

Pattern:

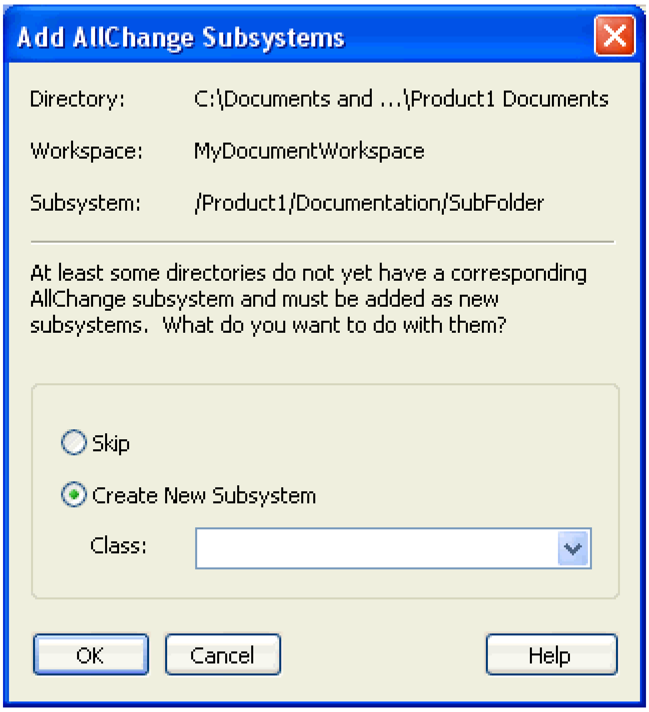
Part State Column: (None)

1 item(s) selected

To check your files into AllChange simply select the files and folders you want and using the right click context menu select **Check In**.

For each folder which does not exist as a **subsystem**[1] beneath the part for the workspace a new subsystem will be created. For each file which does not exist as a component beneath the part for the workspace a new component will be created and the file content stored within AllChange as version 001. The file will be removed from the workspace once it is in AllChange so the Check In operation should be considered as a *move* from the workspace *to* ***AllChange***.

The first time a subsystem is to be created the following dialog will be shown:



Add AllChange Subsystems

Directory: C:\Documents and ...\Product1 Documents

Workspace: MyDocumentWorkspace

Subsystem: /Product1/Documentation/SubFolder

At least some directories do not yet have a corresponding AllChange subsystem and must be added as new subsystems. What do you want to do with them?

○ Skip

● Create New Subsystem

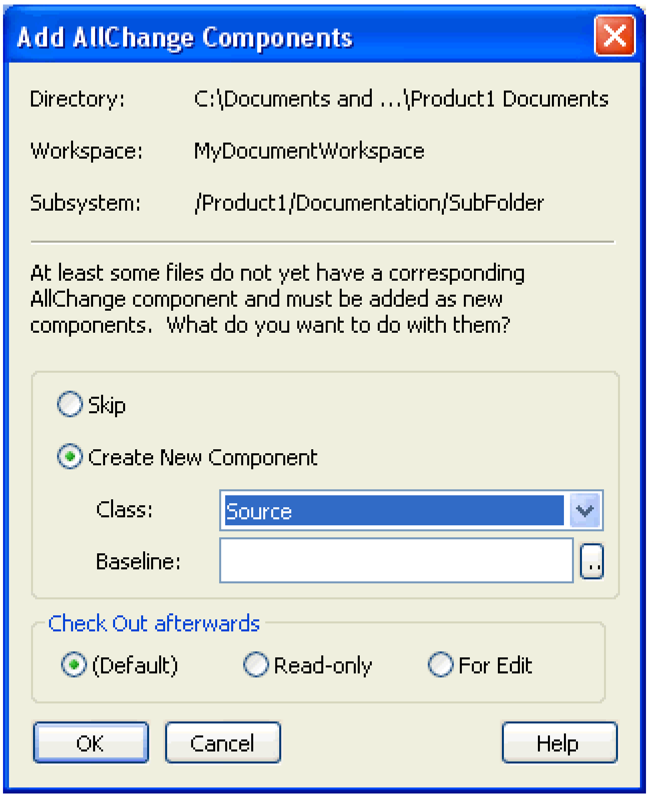
Class:

[1] Subsystem-type parts may have children. Subsystem type is analogous to a directory in an operating system filing system. Note that this term may be mapped to another nomenclature (e.g. Folder)

This verifies that you wish to create subsystems for folders to be checked in and allows you to specify which class to give those subsystems. Any further subsystems created will have this class. As supplied there are no classes for subsystems.

You will also be prompted for file patterns for the files to be checked in. This allows you, for example, to only check in word documents found within the items to be checked in by specifying a pattern of *.doc. If no pattern is specified then all files found will be checked in.

The first time a **component**[1] is to be created for a corresponding file the following dialog will be shown:



Add AllChange Components

Directory: C:\Documents and ...\Product1 Documents

Workspace: MyDocumentWorkspace

Subsystem: /Product1/Documentation/SubFolder

At least some files do not yet have a corresponding AllChange component and must be added as new components. What do you want to do with them?

○ Skip

● Create New Component

Class: Source

Baseline:

Check Out afterwards

● (Default) ○ Read-only ○ For Edit

This verifies that you wish a new component to be created for any files to be checked in and allows you to specify the class that is to be used for those components. The class of a part determines its life-cycle, meta-data associated with that part and other attributes which determine its behaviour. A number of classes are supplied in the Intasoft Standard and ITIL configurations, for details of each of these see Part Classes for Standard Project Template and CI classes for ITIL Configuration respectively.

If you select a class which is *controlled* by a CR (e.g ControlledSource or ControlledDoc) then you will next be prompted to select the CR which authorises this change. Only CRs which are in an appropriate status and assigned to you will be listed. If there are no such CRs then an error will be issued. In this case you may need to progress a CR through its life-cycle to an appropriate status (e.g. InWork) and/or assign it to yourself before you can proceed with the check in, see Using a Change Request to Authorise Changes

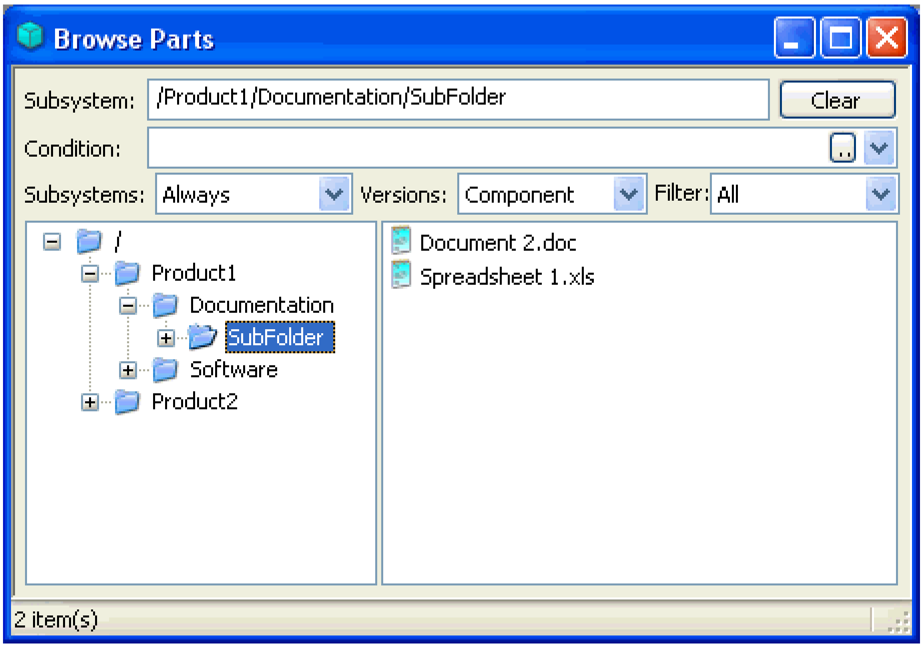
Remember that the term CR may have been redefined (e.g. in the ITIL configuration the term ACMD is used).

Depending on the class you selected for the parts being added to ***AllChange*** there may be additional prompts for meta-data held for that class of part, e.g. for ControlledSource class the Deliverable field will

[1] Component-type parts are at the bottom of the part hierarchy and have no children, but they may have versions. Component type is analogous to a file in an operating system filing system. Note that this term may be mapped to another nomenclature (e.g. Item)

be prompted for (this field is used to determine whether the part/file is to be delivered with the product e.g. usually source code is not delivered but a word document may well be). Meta-data about parts is held in fields on the database record for the part, these are known as *arbitrary fields* and are entirely definable by you per class in your ***AllChange*** project configuration, see Field Name Definitions. For details of the arbitrary fields for the Intasoft Standard configuration see Part Fields for Standard Project Template and for the ITIL configuration see CI Fields for ITIL.

Having completed your check in of your files, if you look in the Parts Browser (CI Browser for ITIL configuration) you will see the files you have checked in in the same folder structure as the directory structure (if any) that you checked in, select the **Parts** (**CIs**) icon in the shortcut bar or **Part | Browse** (**CI | Browse**).



Subsystem: /Product1/Documentation/SubFolder

Condition:

Subsystems: Always

Versions: Component

Filter: All

```
/
  Product1
    Documentation
      SubFolder
    Software
  Product2
```

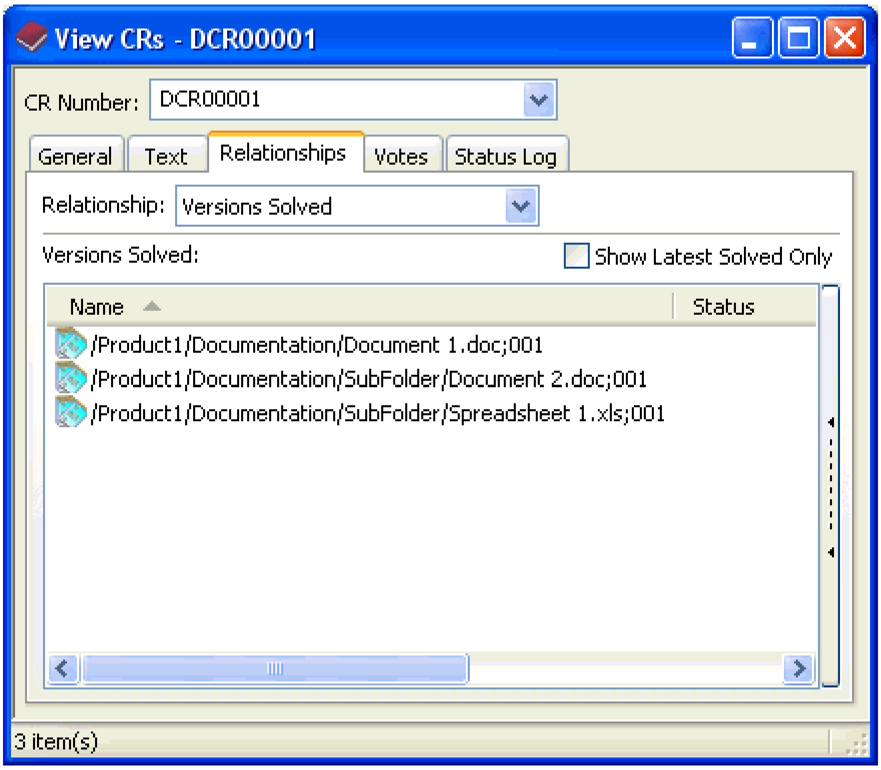
Document 2.doc

Spreadsheet 1.xls

2 item(s)

You will also notice that the files are no longer in the directory from which you checked them in. This is because check in is a *move* operation from the directory to ***AllChange***. If you wish to get the files back into your directory all you need to do is check them out.

If you created your parts of a class which requires a CR (ControlledSource or ControlledDoc) then AllChange will automatically create a link from the CR to the new component (as a part affected) and to the first version of this (as a version solved). This can be seen on the CR viewer **Relationships** tab.



CR Number: DCR00001

Relationship: Versions Solved

Versions Solved: □ Show Latest Solved Only

```
/Product1/Documentation/Document 1.doc;001
/Product1/Documentation/SubFolder/Document 2.doc;001
/Product1/Documentation/SubFolder/Spreadsheet 1.xls;001
```

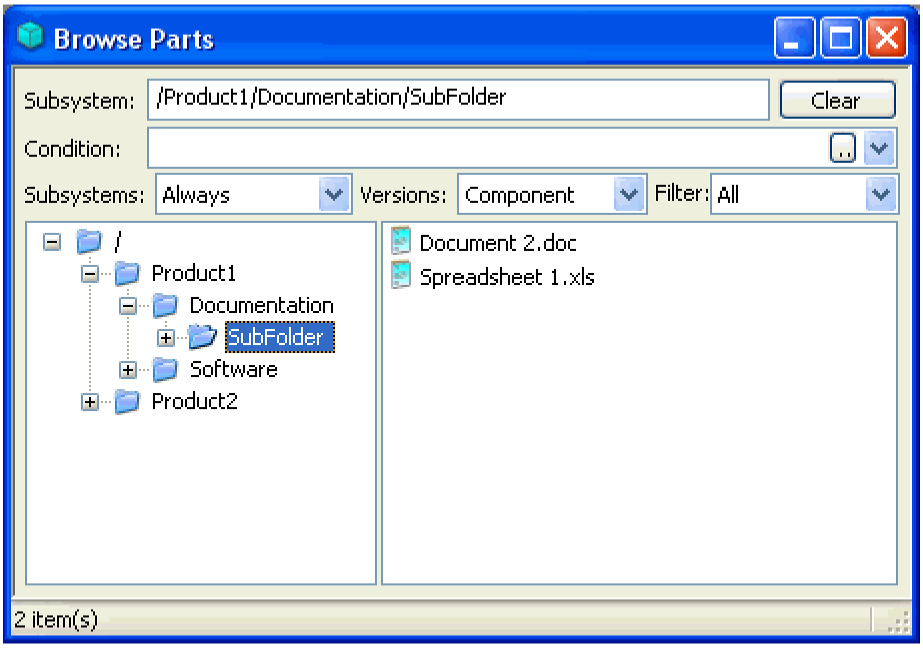
3 item(s)

Parts

Setting up your Configuration Item Structure introduced the concept of parts and Storing some files under AllChange Control took this a bit further and put some files under ***AllChange*** control. Here we look in more detail at parts.

Remember that the term part may have been redefined (e.g. for the ITIL configuration the term CI is used).

In order to see the parts that you have defined to ***AllChange*** you can use the Part Browser (CI Browser for ITIL configuration), this can be opened by selecting the **Parts** icon on the shortcut bar or folder list, or by selecting **Part | Browse**.



Subsystem: /Product1/Documentation/SubFolder

Condition:

Subsystems: Always

Versions: Component

Filter: All

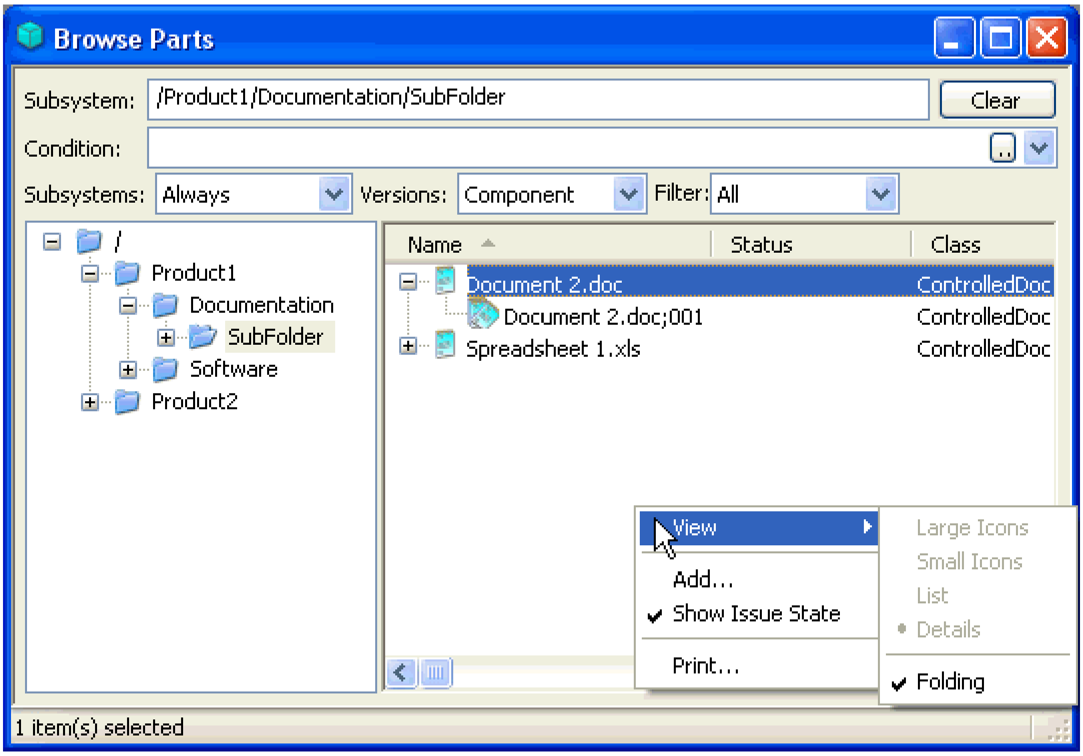
```
/
  Product1
    Documentation
      SubFolder
    Software
  Product2
```

Document 2.doc

Spreadsheet 1.xls

2 item(s)

This allows you to browse around the parts hierarchy. Different views are supported which can be accessed from the right click **View** menu or from **View | View** menu. Amongst others this allows the **Details** of the parts to be shown and also a **Folding** view which allows versions of components to be seen.



Browse Parts

Subsystem: /Product1/Documentation/SubFolder

Condition:

Subsystems: Always

Versions: Component

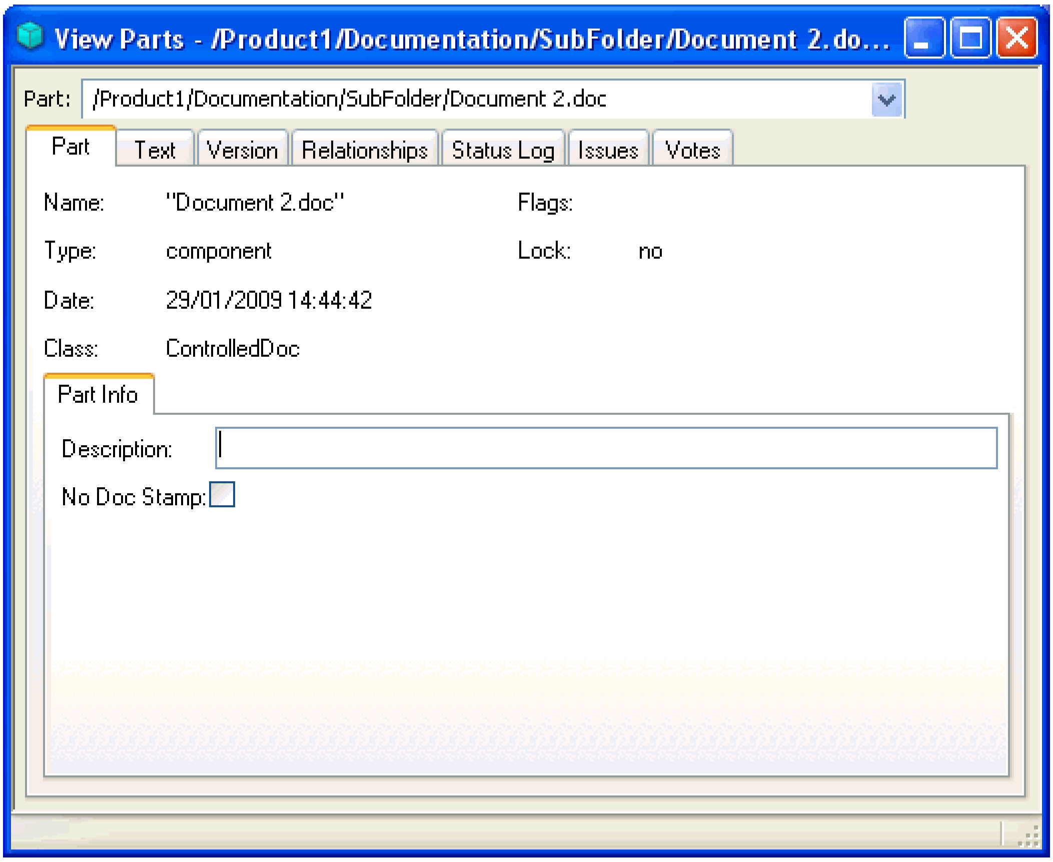
Filter: All

Name	Status	Class
Document 2.doc		ControlledDoc
Document 2.doc;001		ControlledDoc
Spreadsheet 1.xls		ControlledDoc

1 item(s) selected

For full details of the part browser see Browsing the Parts Hierarchy.

In order to view the full details of a component (file), simply double click in the parts browser and the Part Viewer window will be opened showing all information about the component and its relationships to other items in the database.



View Parts - /Product1/Documentation/SubFolder/Document 2.do...

Part: /Product1/Documentation/SubFolder/Document 2.doc

Name: "Document 2.doc"

Flags:

Type: component

Lock: no

Date: 29/01/2009 14:44:42

Class: ControlledDoc

Description:

No Doc Stamp: □

This shows the following information for each part:

- **Name** - The name of the document
- **Type** - the type with be either **component**[1] or **subsystem**[2]
- **Date** - the date that the part was created in AllChange
- **Class** - this defines the **life-cycle**[3], arbitrary fields and Text of the part as well as other attributes such as whether a CR is required in order to make changes to the document. The part shown in the screen above is of class **ControlledDoc** in the Intasoft Standard configuration which requires a CR to make a change and has no life-cycle. Classes are of course entirely definable as required by your organisation, see Classes.
- **Flags** - these will generally be blank, but may show **No File** if it is a part representing hardware/offline items for which there is no corresponding file. The classes **HWComponent** and **ITAsset** in the Intasoft Standard configuration will have the **No File** flag set.
- **Lock** - this indicates whether the part is in a released baseline and may not be deleted

The remainder of the fields are arbitrary fields which vary according to the class of part. Up to 40 of these may be defined per class and placed on any named tab of your choice, see Field Name Definitions. Those shown in the screen above are those defined for the ControlledDoc class of component in the Intasoft Standard project configuration.

Fields which may be changed using controls on the part viewer may be modified as required and then select **Misc | Update** or the 💾 toolbar button to save your changes.

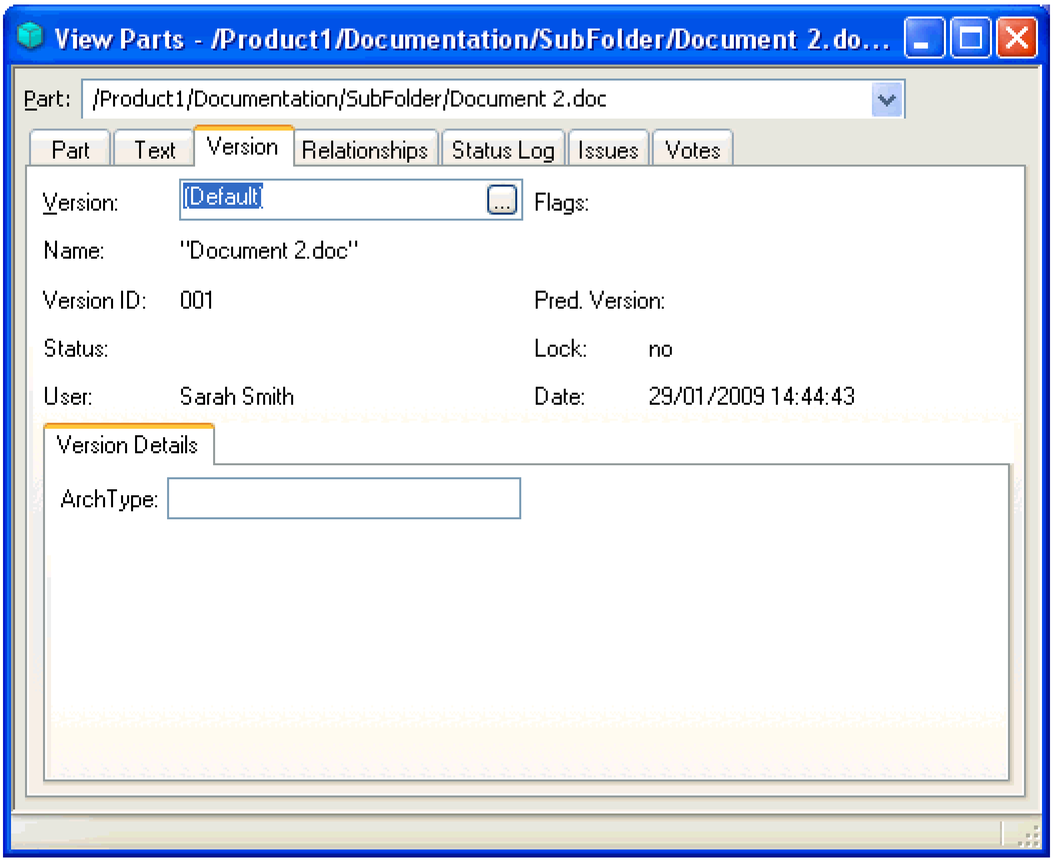
The **Text** tab allows arbitrary descriptive text to be associated with the component and/or each version of the component.

For each component ***AllChange*** maintains a complete version history. The **Version** tab shows information about a specifically selected version:

[1] Component-type parts are at the bottom of the part hierarchy and have no children, but they may have versions. Component type is analogous to a file in an operating system filing system. Note that this term may be mapped to another nomenclature (e.g. Item)

[2] Subsystem-type parts may have children. Subsystem type is analogous to a directory in an operating system filing system. Note that this term may be mapped to another nomenclature (e.g. Folder)

[3] A life-cycle (or simply cycle) is defined as a series of statuses through which a part, CR or baseline passes and may be used to control access to items and implement approval procedures etc.



View Parts - /Product1/Documentation/SubFolder/Document 2.do...
Part | Text | Version | Relationships | Status Log | Issues | Votes
Version Details

Part: /Product1/Documentation/SubFolder/Document 2.doc

Version: (Default)

Flags:

Name: "Document 2.doc"

Version ID: 001

Pred. Version:

Status:

Lock: no

User: Sarah Smith

Date: 29/01/2009 14:44:43

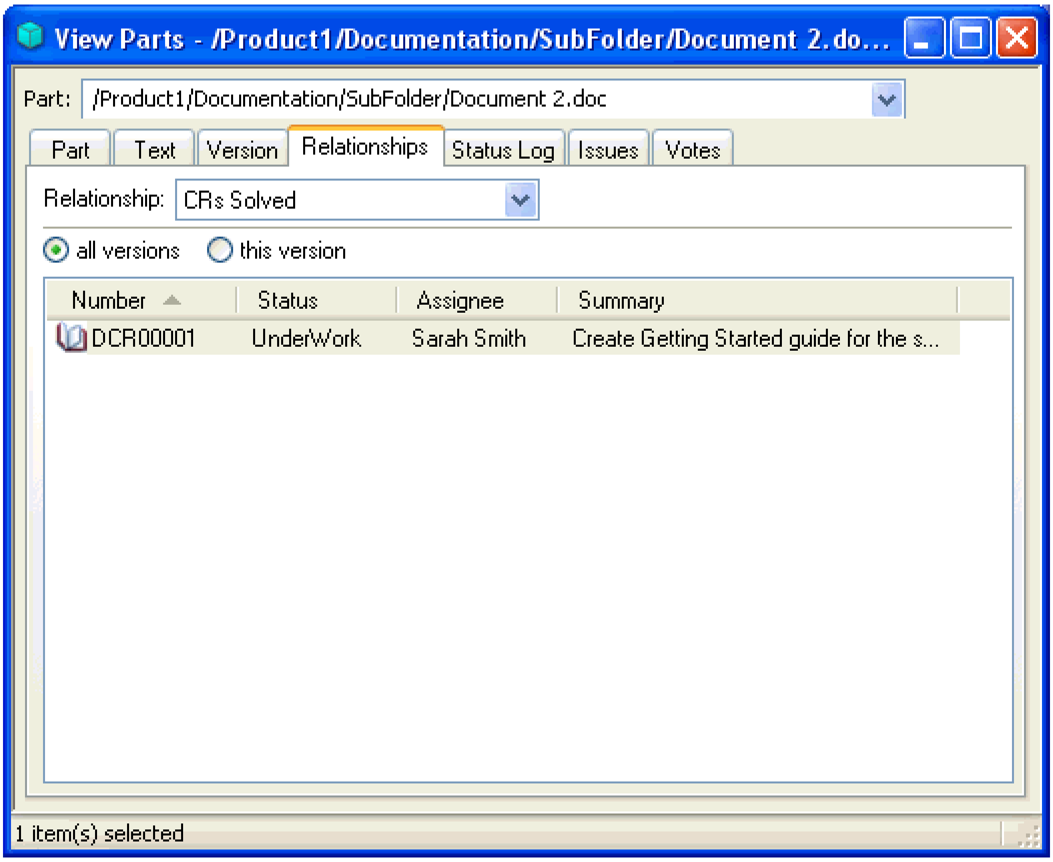
ArchType:

The information shown includes:

- **Version ID** - this will be a 3 digit number by default.
- **User** - who created this version
- **Date** - on which this version was created
- **Status** - the stage in the life-cycle for this version. Note that this may be blank if there is no life-cycle for the class of part,

In addition there are arbitrary fields which vary according to the class of part. As with the component arbitrary field, up to 40 of these may be defined per class (for each version) and placed on any named tab of your choice, see Field Name Definitions.

On the **Relationships** Tab we can see the relationships to other items such as the CR authorising the change.



View Parts - /Product1/Documentation/SubFolder/Document 2.do...

Part: /Product1/Documentation/SubFolder/Document 2.doc

Part
Text
Version
Relationships
Status Log
Issues
Votes

Relationship: CRs Solved

● all versions ○ this version

Number	Status	Assignee	Summary
DCR00001	UnderWork	Sarah Smith	Create Getting Started guide for the s...

1 item(s) selected

.

Changing a file

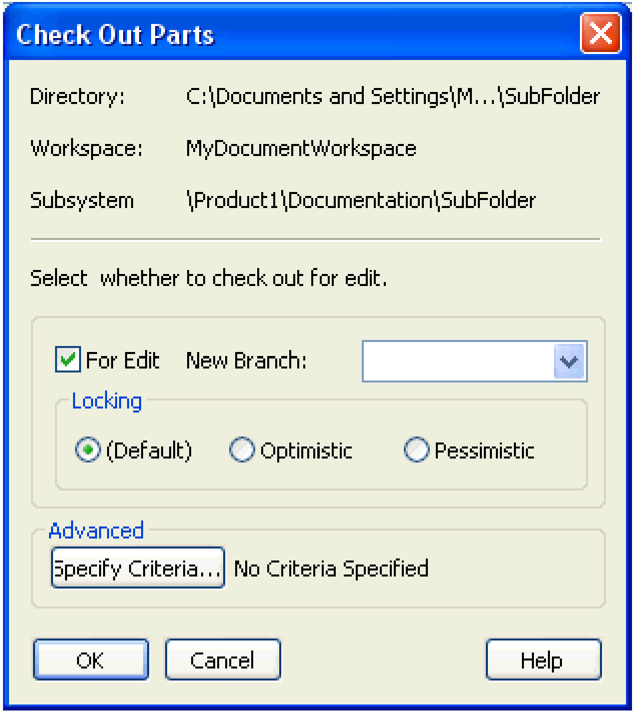
Having put some files under AllChange control you will probably want to make changes to these at some time.

Storing some files under AllChange Control introduced the concept of a workspace and *checking files in* to ***AllChange***. A workspace is a location on your PC or network where you wish to have working copies of files. In order to make a change to a file we *check it out to a workspace*. Also informing ***AllChange*** that the purpose of this is to make a change (**Edit**) the file.

Furthermore, if the part is of a class which is controlled by CRs for changes (e.g. ControlledSource or ControlledDoc in the Intasoft Standard configuration), then a CR will be required to authorise the change. This will need to be assigned to you and in the appropriate status for making a change (e.g. UnderWork), see Using a Change Request to Authorise Changes. If you require a CR for change then ensure that you have a suitable one before attempting to make your change,

There are many different ways of checking a file out to make a change, but one of the basic ones is as follows:

1. Attach to the workspace where you wish the file to placed, use **Workspace | Attach Workspace**.
2. Select the part(s) you wish to check out using the Part Browser, use **Part | Browse**.
3. Select Check Out (right click context menu or **Part | Check Out**)



Check Out Parts

Directory: C:\Documents and Settings\M...\SubFolder

Workspace: MyDocumentWorkspace

Subsystem \Product1\Documentation\SubFolder

Select whether to check out for edit.

☑ For Edit New Branch:

Locking

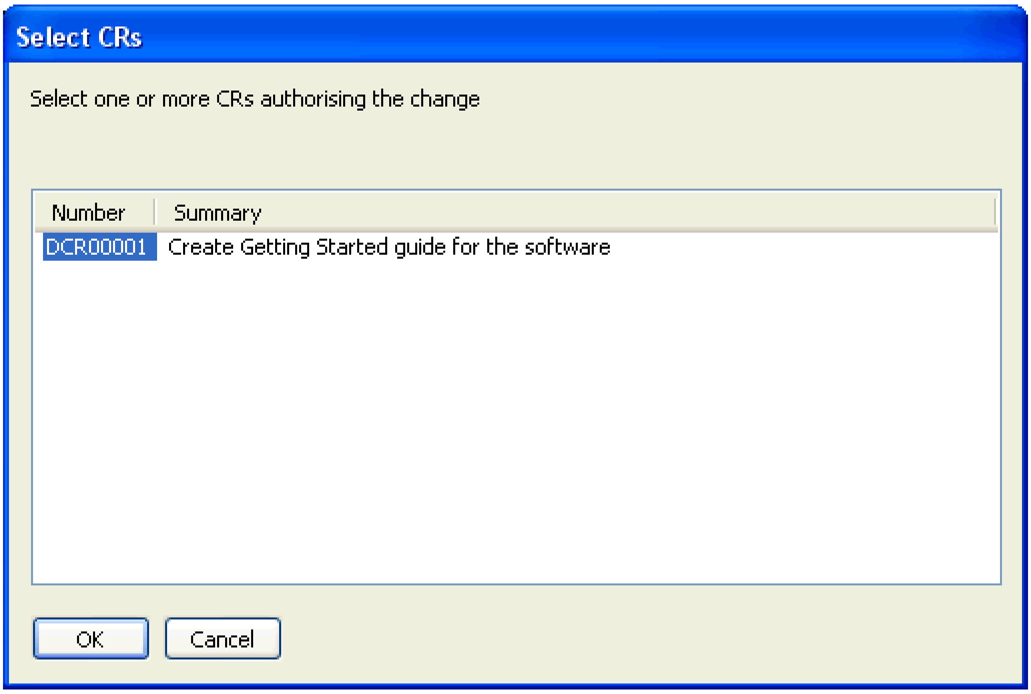
● (Default) ○ Optimistic ○ Pessimistic

Advanced

Specify Criteria... No Criteria Specified

Select the **For Edit** check box and click **OK**.

4. If your part requires a CR in order to make changes you will then be prompted for the CR which authorises this change,



Select one or more CRs authorising the change

Number	Summary
DCR00001	Create Getting Started guide for the software

If you have no CRs in an appropriate status/assignment then an error will be given (There are no CRs which are valid for editing this part). Select the CR which authorises this change and **OK**.

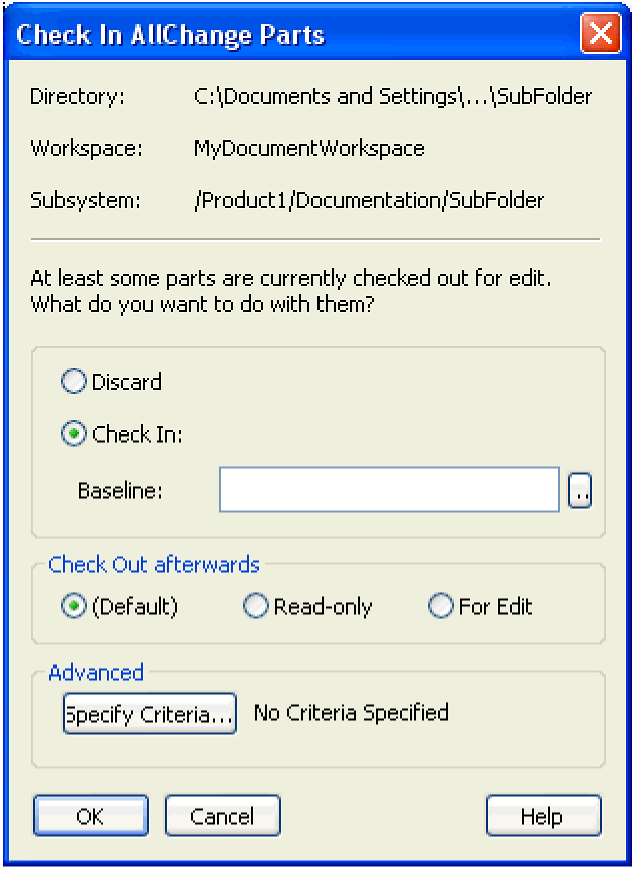
Your part(s) are now checked out for edit purposes to your workspace. You can look in the file browser or use Windows Explorer or any other tool to see the files in your workspace directory. You can now modify the files using an appropriate tool (e.g. MS Word, Visual Studio etc depending on the type of file).

Note that AllChange has recorded the fact that you have this file for editing purposes and will record a new version of it so that you are working in a copy and previous versions are never lost or overwritten. See Checking Files in and out of AllChange for full details.

In order to store your changes back in ***AllChange*** as new version of the component part, you then check them in, just as we did when adding new files into ***AllChange***. The basic steps are as follows:

1. Attach to the workspace where the files to be checked in are located if you are not already attached, use **Workspace | Attach Workspace**.
2. Select the part(s)/file(s) you wish to check in using the Part Browser, use **Part | Browse** or File Browser (**File | Browse**).

3. Select Check In (right click context menu or **Part | Check In**)



Check In AllChange Parts

Directory: C:\Documents and Settings\...\SubFolder

Workspace: MyDocumentWorkspace

Subsystem: /Product1/Documentation/SubFolder

At least some parts are currently checked out for edit.
What do you want to do with them?

○ Discard

● Check In:

Baseline:

Check Out afterwards

● (Default) ○ Read-only ○ For Edit

Advanced

No Criteria Specified

Click **OK**.

If the part is of class **Document** or **Source** then you will be prompted for a Comment. This is a comment describing the change/reason for change and will be stored with other information about the new version checked in. If the part is of class ControlledSource or ControlledDoc, for example, no comment is requested as the CR authorising the change can describe the change adequately. This is the behaviour in the out-of-the-box configurations but may be tailored as required.

The new version of the file has now been stored in ***AllChange*** together with Date/Time/Author and CR authorising change information and may be retrieved again at any time.

This illustrates the basic method of checking out and checking in from within ***AllChange***, but there are a number of interfaces to various tools which also allow files to be checked in and out from those tools. This may be a preferred method depending on the tool being used. Examples of interfaces include:

- **Windows Explorer** - Using Right Click menus you can perform various ***AllChange*** operations directly on files and folders.
- **MS Word/Excel** - Check In/Out may be performed within MS Word and Excel by an **AllChange** menu which is present when the interface is installed
- **MS Visual Studio** - ***AllChange*** supports the Microsoft Common Source Code Control Interface (MCSCCI) and can be used from within environments such as Visual Studio which support this interface
- **Eclipse** - files may be checked in and out and updated from the Team menu in Eclipse

You may have noticed on the **Check Out** dialog an option to select **Pessimistic** or **Optimistic** Locking. These are the two models that ***AllChange*** supports to prevent multiple users making changes at the same time and then overwriting each others changes. The default is pessimistic locking in the supplied configurations although certain interfaces will use the optimistic model (e.g. Eclipse). For details of the 2 models see Checking Files In and Out of AllChange.

Creating Parts with no file

So far in looking at configuration items (parts) we have only been looking at the need to control files which is useful for software development, documents and any other electronic document. However, ***AllChange*** may also be used to hold information about other configuration items for which there may not be an electronic document for example IT Assets or components in a manufacturing scenario. These are represented as parts with no file, other than that they can behave in exactly the same way as parts which do have a file.

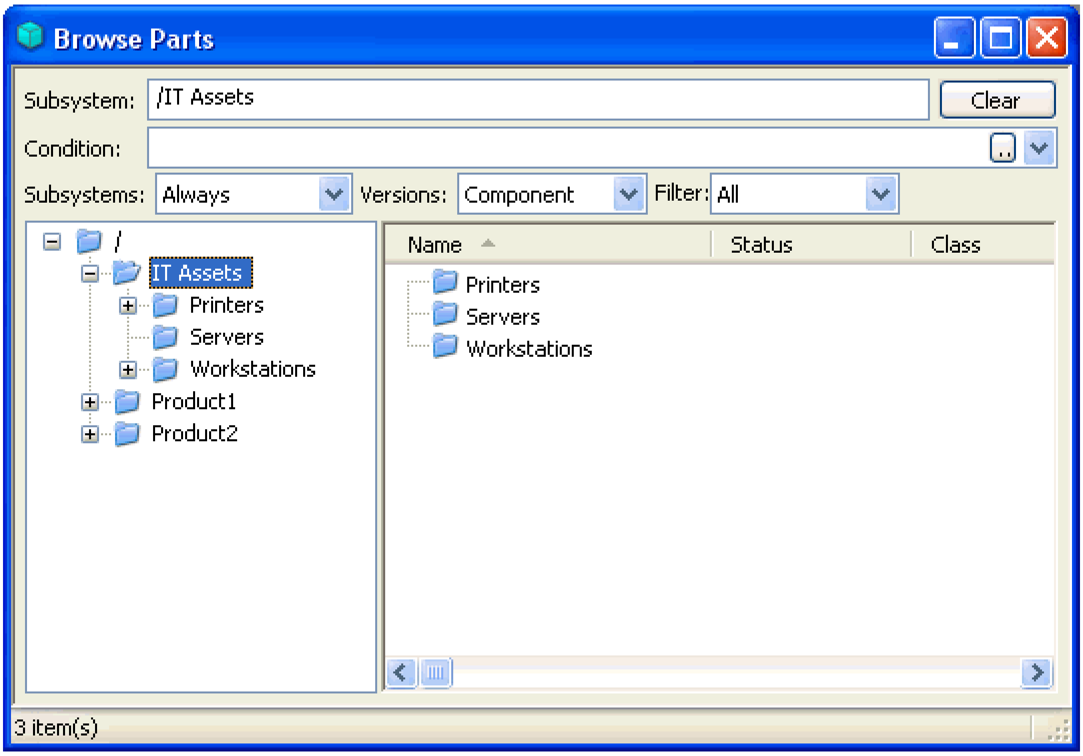
Two classes for components are supplied in the Intasoft Standard and ITIL project configurations for this purpose:

- **ITAsset** - this is intended for use with configuration items used to represent IT Assets, for example, Servers, Workstations, Printers etc.
- **HWComponent** - this is intended for use for a more generic hardware model including manufacturing components.

Remember that in the ITIL configuration parts are referred to as CIs.

Both of these classes are controlled by CRs (i.e. they require a CR to authorise any change or their addition to the database). In the Intasoft Standard configuration the ITAsset class parts are controlled by ITCR class CRs whereas HWComponent class parts are controlled by ECR (Engineering Change Request) class CRs. These classes are provided as samples only but they may provide you with a template to modify to your own specific requirements. Note also that you may create your own classes or modify those supplied such that parts of that class are not controlled by CRs if this is not required.

To create an IT Asset Register or a product manufacturing model first you must decide on the structure within the hierarchical parts tree that would best model your requirements and create this hierarchy as in Setting up your Configuration Item Structure. The example shown below might be suitable for an IT Asset Register.



Subsystem: /IT Assets

Subsystems: Always

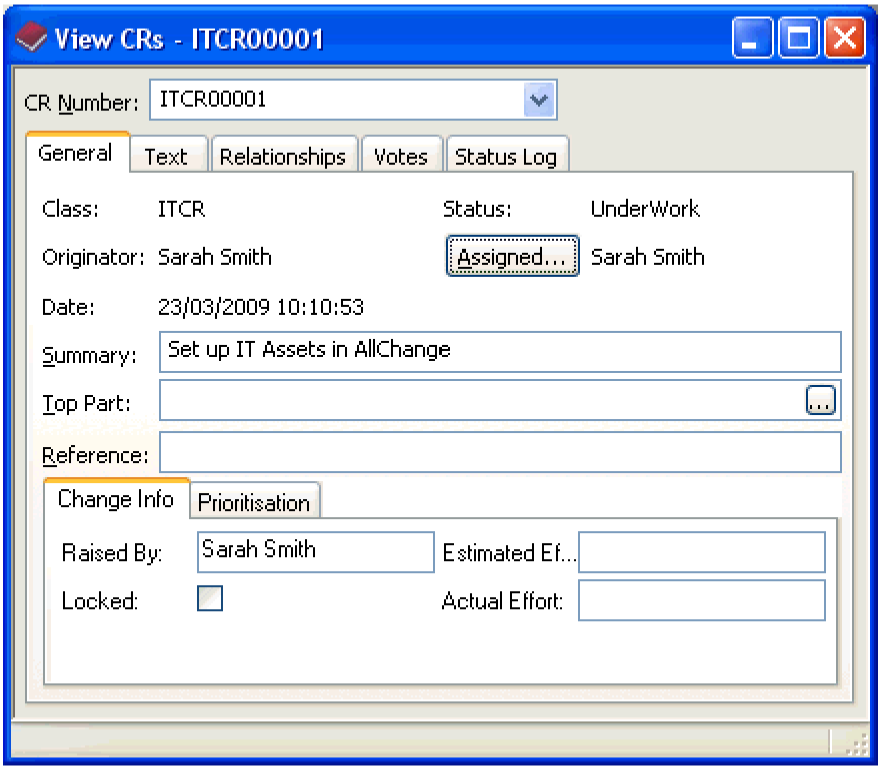
Versions: Component

Filter: All

3 item(s)

You will need a CR to authorise the addition of ITAsset class components into the folder structure. For the Intasoft Standard configuration this should be of class ITAsset and it must be assigned to you and in the

UnderWork status. For the ITIL configuration (where CRs are referred to as ACMDs) this should be an RFC, assigned to you and in status *Approved*, see Using a Change Request to Authorise Changes.



View CRs - ITCR00001

CR Number: ITCR00001

Class: ITCR

Status: UnderWork

Originator: Sarah Smith

Assigned... Sarah Smith

Date: 23/03/2009 10:10:53

Summary: Set up IT Assets in AllChange

Top Part:

Reference:

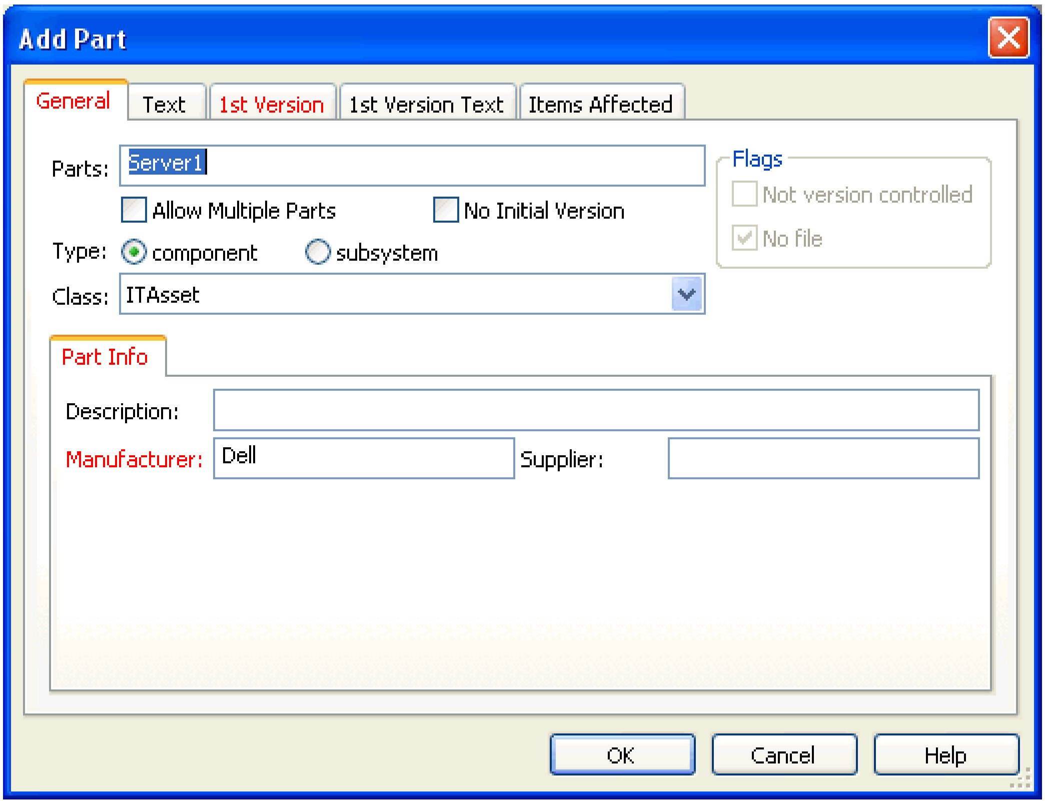
Raised By: Sarah Smith

Estimated Ef...

Locked: □

Actual Effort:

You can then add component parts of class ITAsset into the appropriate subsystems, select **Part | Add** and select the class ITAsset, notice that the **No File** flag is checked, this is an attribute of the class:



Add Part

Parts: Server1

□ Allow Multiple Parts

□ No Initial Version

Type: ● component ○ subsystem

Class: ITAsset

Flags

□ Not version controlled

☑ No file

Description:

Manufacturer: Dell

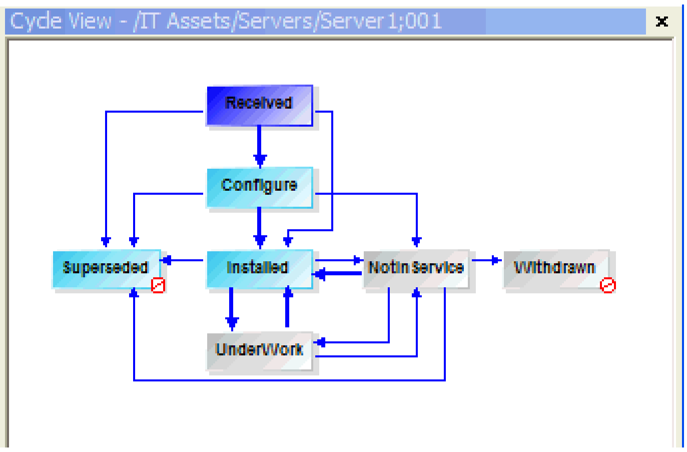
Supplier:

The class of ITAsset as supplied has some example arbitrary fields defined, Description, Manufacturer and Supplier for the part, and Notes, Location and Cost for each version. Of these the Manufacturer and

Notes on the first version are compulsory when the part is created as indicated by the fields in red. Up to 40 arbitrary fields may be defined so a lot of additional information can be stored about each asset as required, see Field Name Definitions. Once you have clicked Ok to create the component, you will be prompted to select the CR which authorises this change. For the Intasoft Standard configuration this should be an ITCR and for ITIL it will be an RFC.

You can repeat this process as often as required to add more parts as needed.

Having created some components to reflect your IT Assets you will notice that this class of part has a life cycle associated with it.



Parts can have a life-cycle associated with them in the same way as CRs do and can vary according to the class. The HWComponent class of part, for example, has no life-cycle. This is all configurable according to your requirements and the cycles supplied are just examples which may help to get you started. The ITAsset cycle provided can be used to show when an item which has been ordered has been received, when it is being configured and then installed, for full details of the life cycle see **ITAsset Cycle Driven Change Process** in Part Change Processes. To progress the part through its life-cycle you can simply double click on the required status in the cycle viewer window.

You have now reached the end of the Getting Started Guide. You have been seen how to install ***AllChange*** and set up an initial project. You have been introduced to the Change Control, Configuration Management (of both files and non file based items), Version Control and life-cycle processes in ***AllChange***. You have also been shown how flexible ***AllChange*** is allowing it to be tailored to your organisations processes and procedures. There are many more features and facilities all of which are documented in the user manuals. For support please do not hesitate to contact us (support@intasoft.net, or via the web site - www.intasoft.net). Further resources including a support forum are also available in the members area of the website which is accessible to all customers with a support contract, to register visit http://www.intasoft.net/passreg.asp

Index