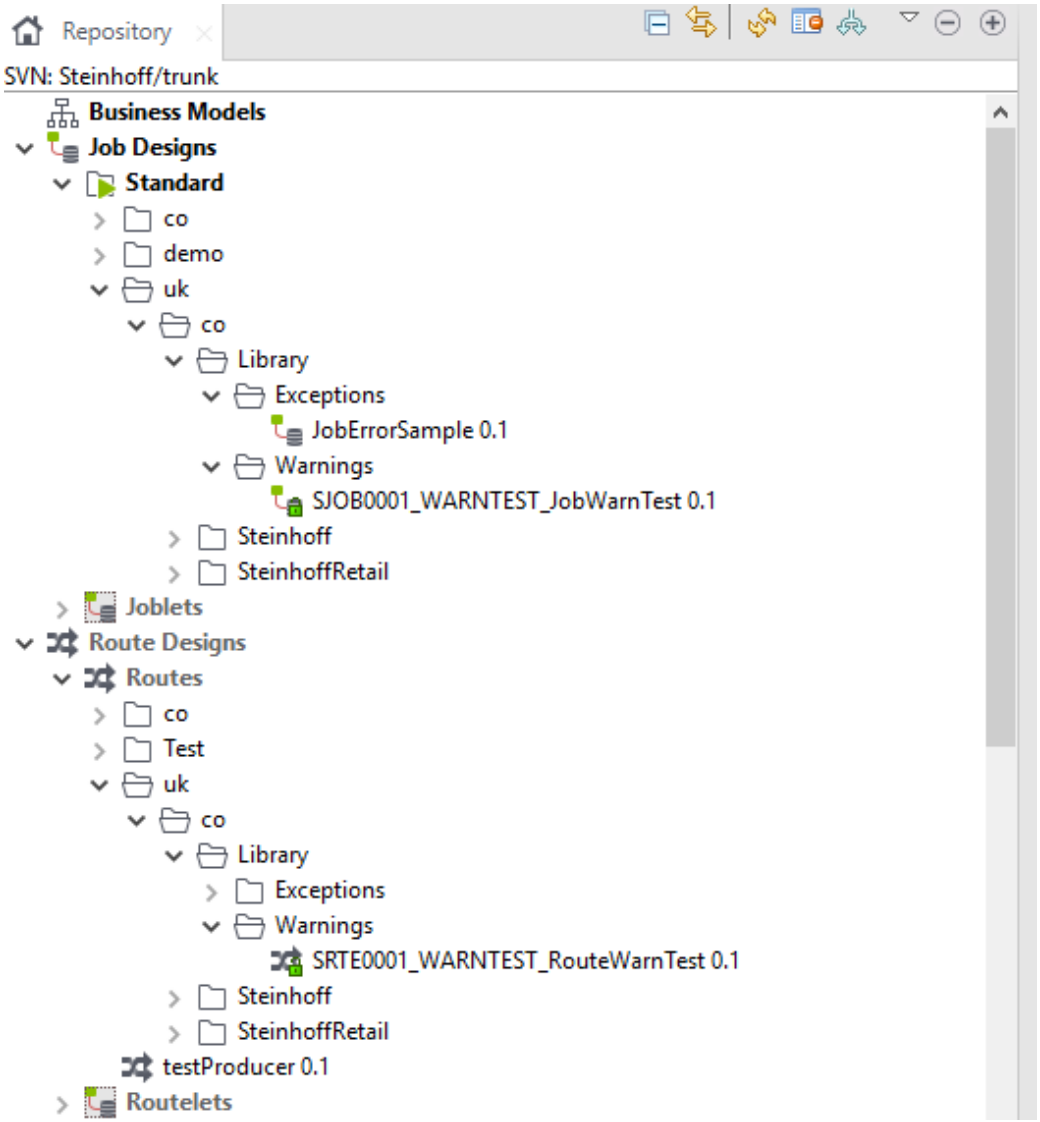
Warnings

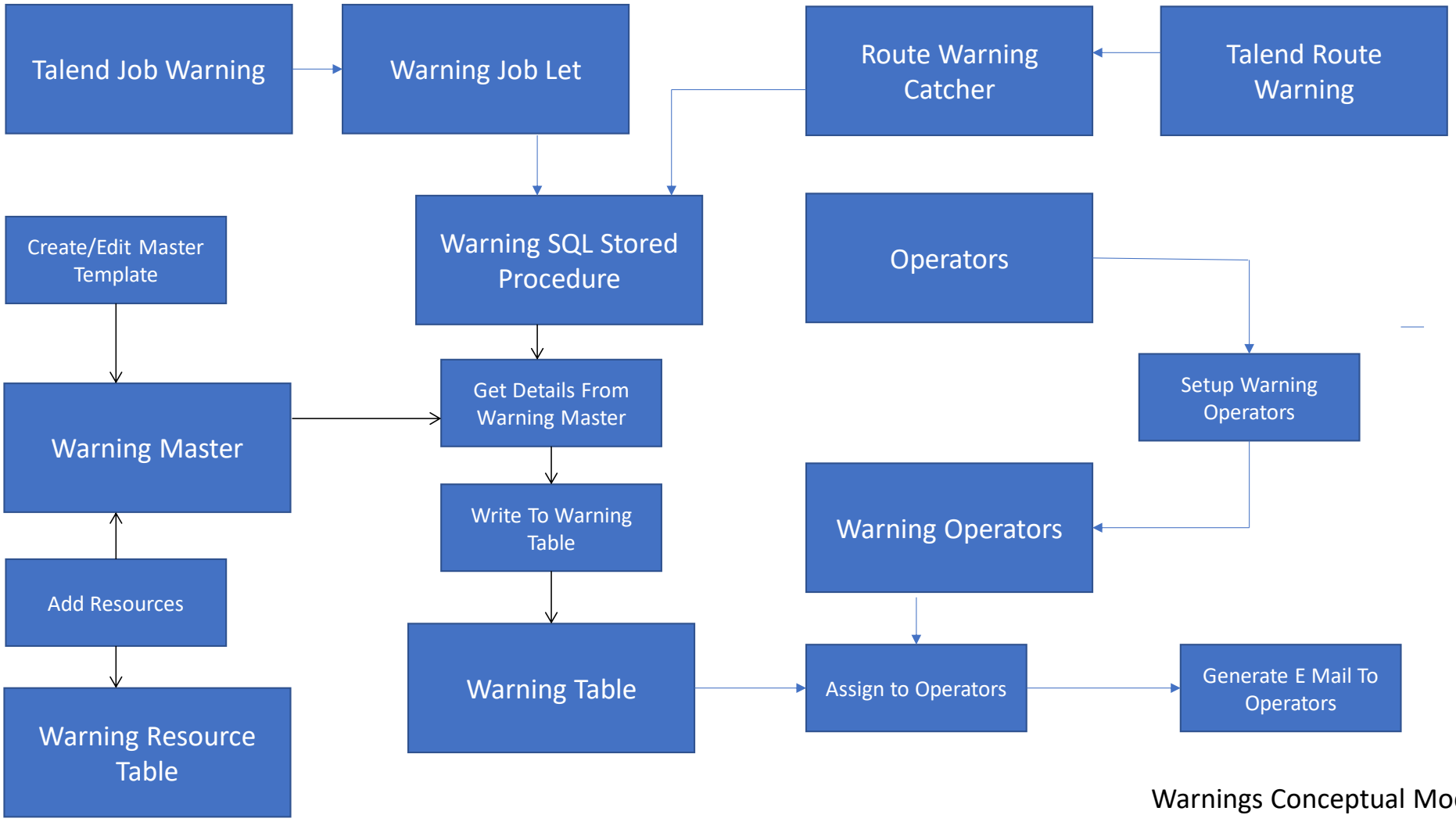
Overview

Warnings are intentionally raised messages to notify interested parties when a certain condition has been reached within a process that will require their attention. A typical scenario would be the absence of a data file preventing further processing. Classified into 2 groups, a standard warning will issue a message and resume processing whilst a critical warning will terminate the process after sending the notification.

Handled using a similar mechanism to exceptions with the main difference being that these are all known scenarios catered for within the program logic. There are drop in components within Talend to handle the occurrence of warnings. Demonstration of these together with the components used to raise the warnings can be found for both jobs and routes within the SVN: Steinhoff project accessed via Talend Open Studio (TOS).



The concept diagram for the warning system is shown below:



Warnings Conceptual Model

1. Setting up a warning

The following terms will be used henceforth to describe the processes and tools used:

TOS	Talend Open Studio
Framework	The error and warning framework application
Standard	A standard warning that reports but continues processing
Critical	A warning that terminates processing after notification
Operator	Person(s) set up to receive the warning notification by e-mail
Resource	An external document used to provide additional insight into warning conditions
Master	A template containing details of the warning

Setting up a warning consists of the following steps:

- Create the warning conditions in the interface using TOS
- Add the warning handling components in TOS
- Set up the master within framework
- Assign operators to the master
- Optionally add resources to the master

Once correctly set up reaching the warning conditions within the Talend interface will cause a warning to be generated which will then be captured and loaded into the framework database. Data from the master will be used to generate an e-mail notification that is sent to the operator(s).

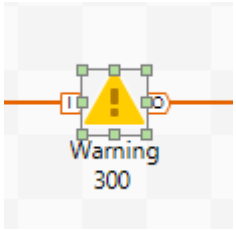
Each step will now be examined in detail.

1.1 Create warning conditions in the interface using TOS

This and the following section will describe the process for Talend jobs first, followed by routes.

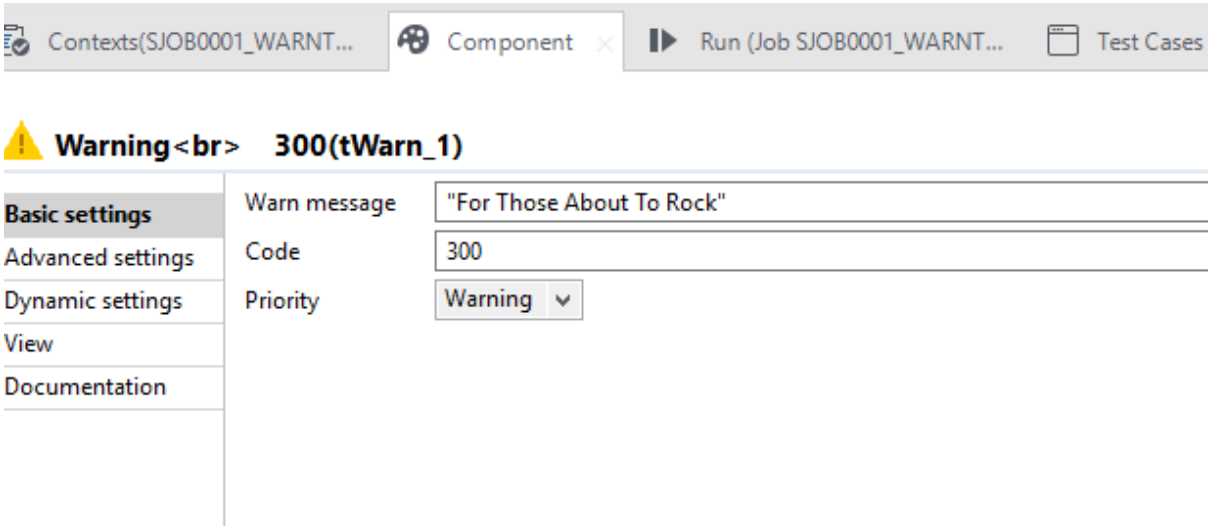
1.1.1 Talend Job Warnings

Standard warnings are generated using the tWarn component placed at the appropriate point within the job. A suggestion is that the label should be configured to display the fact it is a warning and show the error code as below.



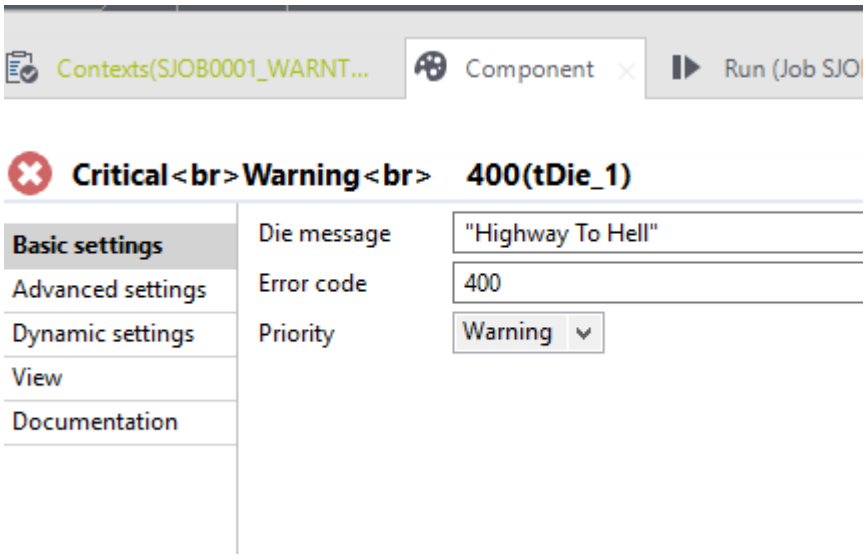
Configuration of the component includes a code and a message. The code is shown as numeric in this example but will accept alphanumeric values. The value needs to be unique and this is maintained in the framework via the master template. Note that warnings will not be loaded into the framework until a master for the code has been created.

The warning message should just be a short reference to what the warning is for, full details are held in the template. Priority should be left as 'Warning'.



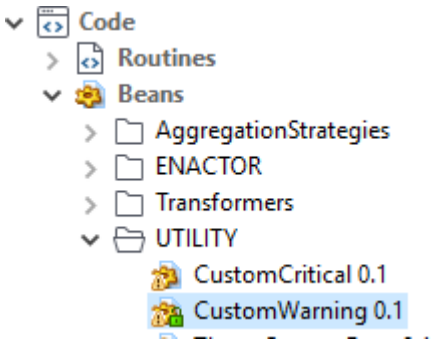
Critical warnings use the tDie component instead which will terminate processing after issuing the warning. The component label should be configured in the same way as warnings

Configuration of the component is identical to standard warnings and all the same caveats apply.



1.1.2 Route Warnings

Talend routes use Apache Camel components and there is no standard equivalent to the warning components used by jobs. To represent warnings a couple of java beans have been created to extend the standard exception object with the addition of a string property. These beans can be found in the code section of SVN displayed in TOS.

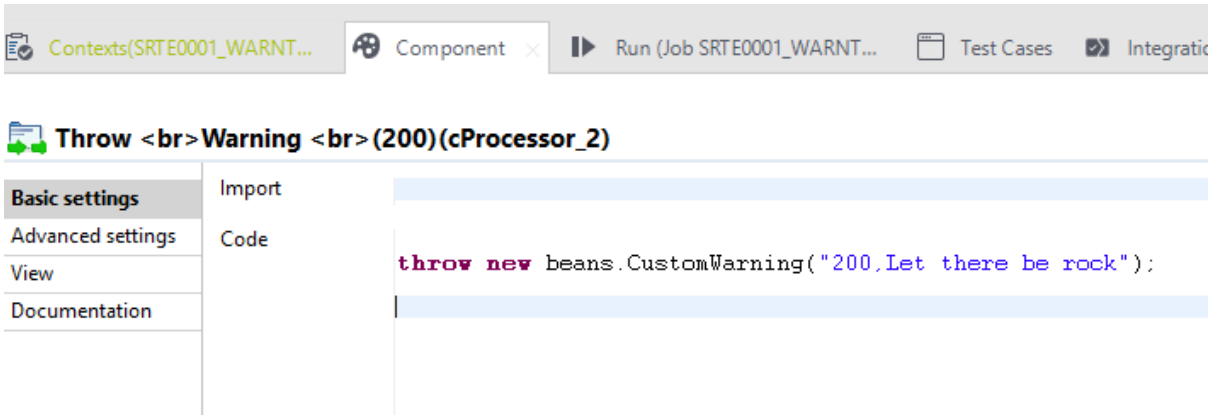


The beans are then called within the route using a cProcessor component.

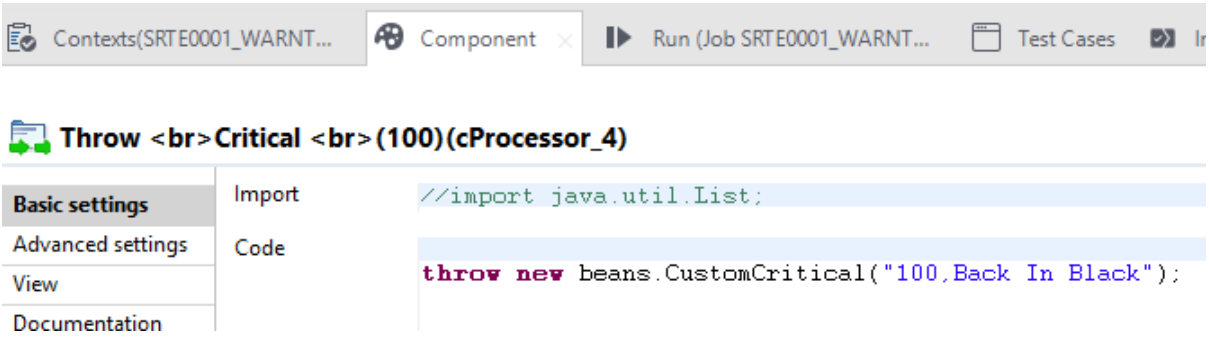
Standard warnings are raised using a cProcessor to raise an instance of the CustomWarning bean. Since this is just an extension of the exception class it can be raised in the same way with additional information relevant to the warning stored in the string property added by the extension.

As with jobs the label should be formatted to indicate its purpose and the error code.

Unlike the job components there is only a single string parameter so this should take the format <error code>,<message> again the code can be alpha numeric if required. The routine that picks up the warning will take responsibility for splitting the string into separate code and description values.



Critical errors work in the same way but use CustomCritical bean instead



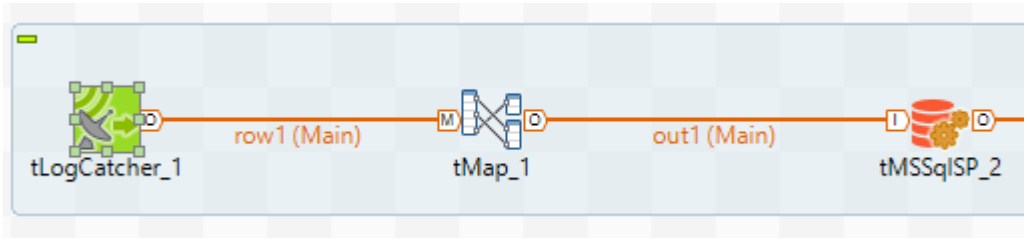
1.2. Add the warning handling components in TOS

The warning handling components for both jobs and routes work in a very similar way to the exception versions although they are kept separate as not all interfaces require warnings to be handled. It is assumed that any interface using warnings will already have the exception components in place so configuration values will be handled by default.

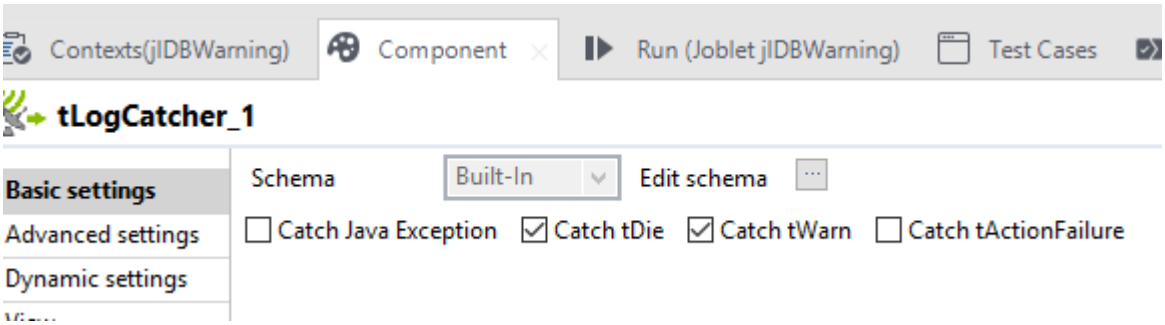
1.2.1 Job Warning Handling

Warnings are handled simply by dropping the joblet jlDBWarning into the job

If the joblet is exploded it is shown to contain the following 3 components:



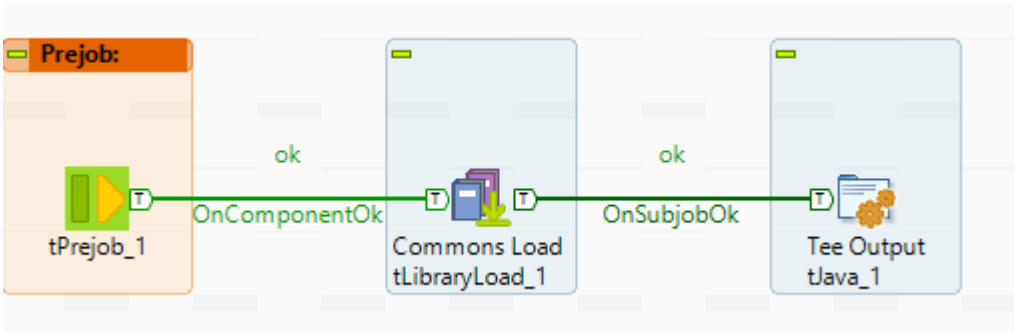
The log catcher is configured to catch items raised by the tWarn and tDie components:



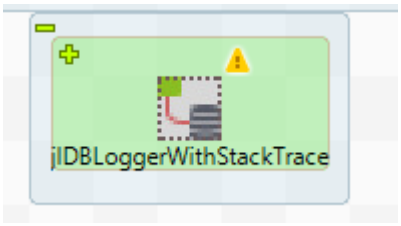
The tMap converts the data captured by the log catcher into the input parameters for the SQL stored procedure called by the tMSSqlSP_2 component, which loads the warning data into the framework database provided a master template for the error code has previously

been set up. Otherwise the data is ignored and there is no concept of an unknown warning as these will all have been user defined.

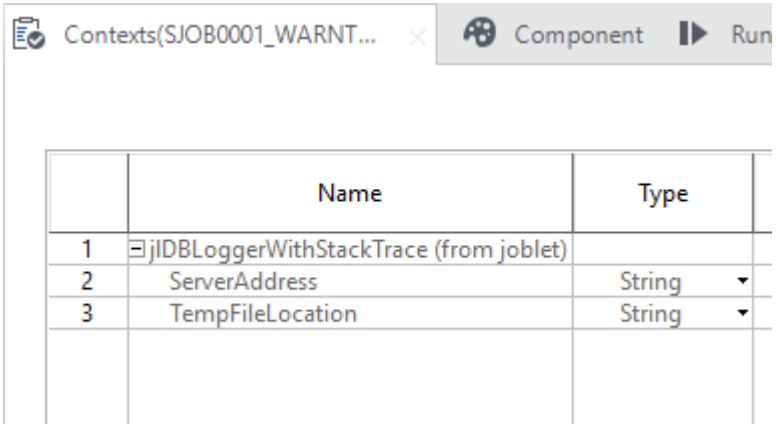
Standard and Critical types are both handled by the same components. The location of the database server is determined by the context variable loaded in the exception routines, so it is essential these components are also part of the job. For reference these are:



For the library pre load



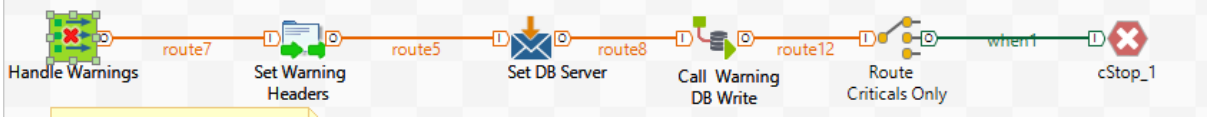
And the exception joblet. Adding these will automatically add the following data to the context list:



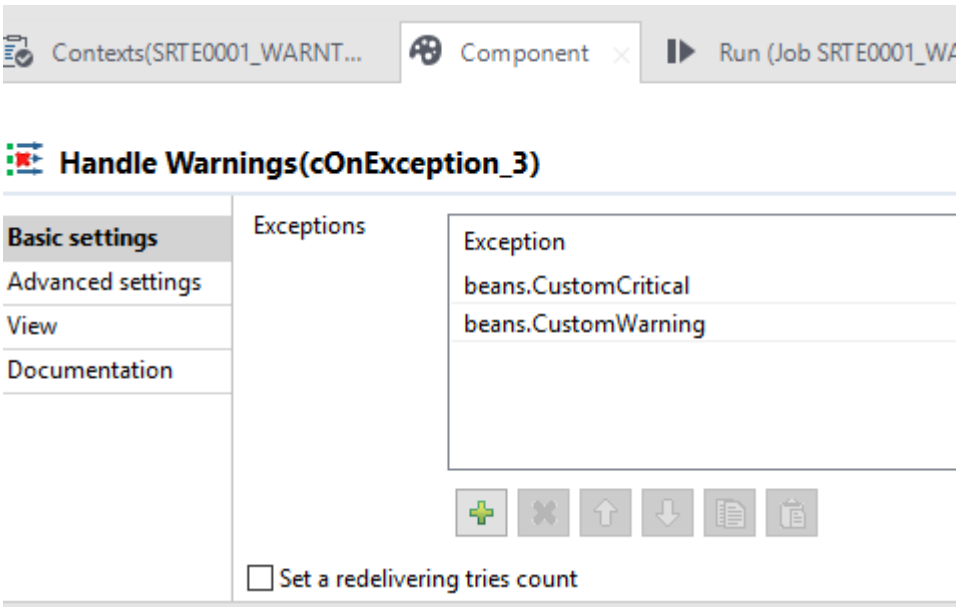
The ServerAddress value is used by the warning mechanism to locate the framework database server

1.2.2 Route Warning Handling

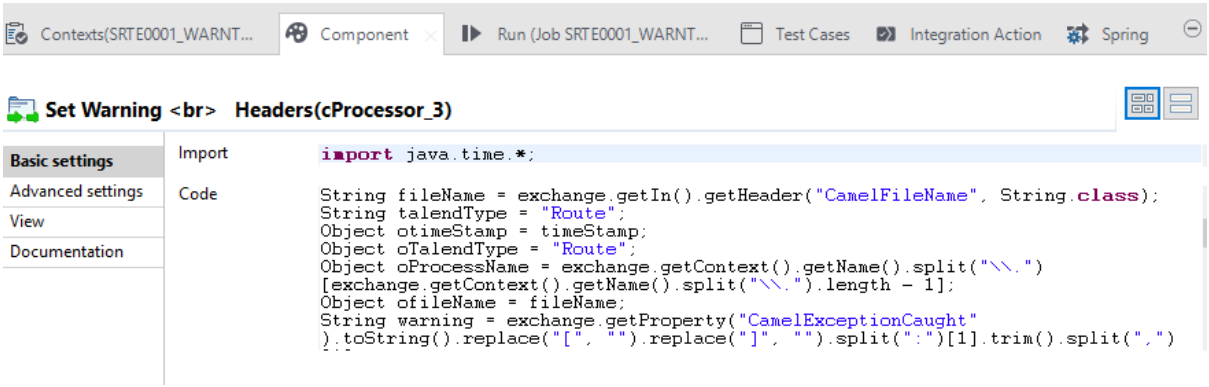
Warnings are handled in routes by dropping the following set of components into it.



Handle Warnings is a cOnException component configure to capture exceptions raised as instances of beans.CustomCritical or beans.CustomWarning as defined in the previous section.

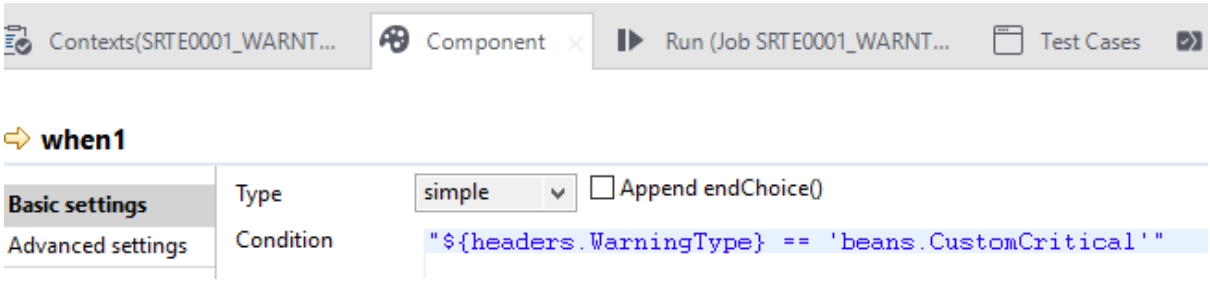


Set Warning Headers is a cProcessor component that injects a java snippet into the route. This code extracts data from the exception caught be previous component and transforms it into a new format. As part of this process the string entered into the warning components in the route is split into code and description. The data is written into the message header for further processing.



SetDBServer picks up the database server address from the context which would already be in place from the error components and writes this value into the message header. A small job that runs the same SQL stored procedure used by the warning handling for jobs is then called with the Call Warning DB Write component. This loads the warning data into the framework database provided a warning master has been previously set up.

After loading a check is made to see which type of bean generated the warning and if it is a critical type then call the cStop component to prevent further processing of the interface.



1.3 Set up the master within framework

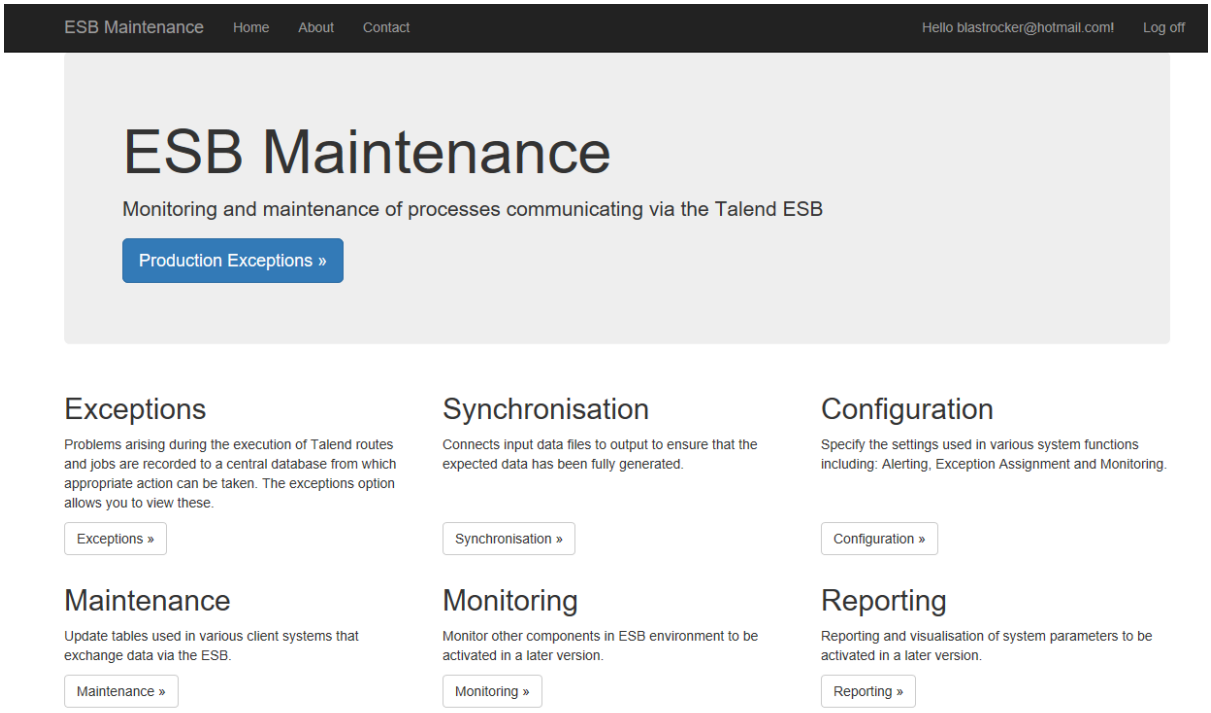
The framework consists of a central SQL Server database which records the exception and warning information fed to it together with various configuration tables and a web application used to administer and retrieve this information. The application can be located here:

http://EVDCSTEMDB02/DBTablemaintenance

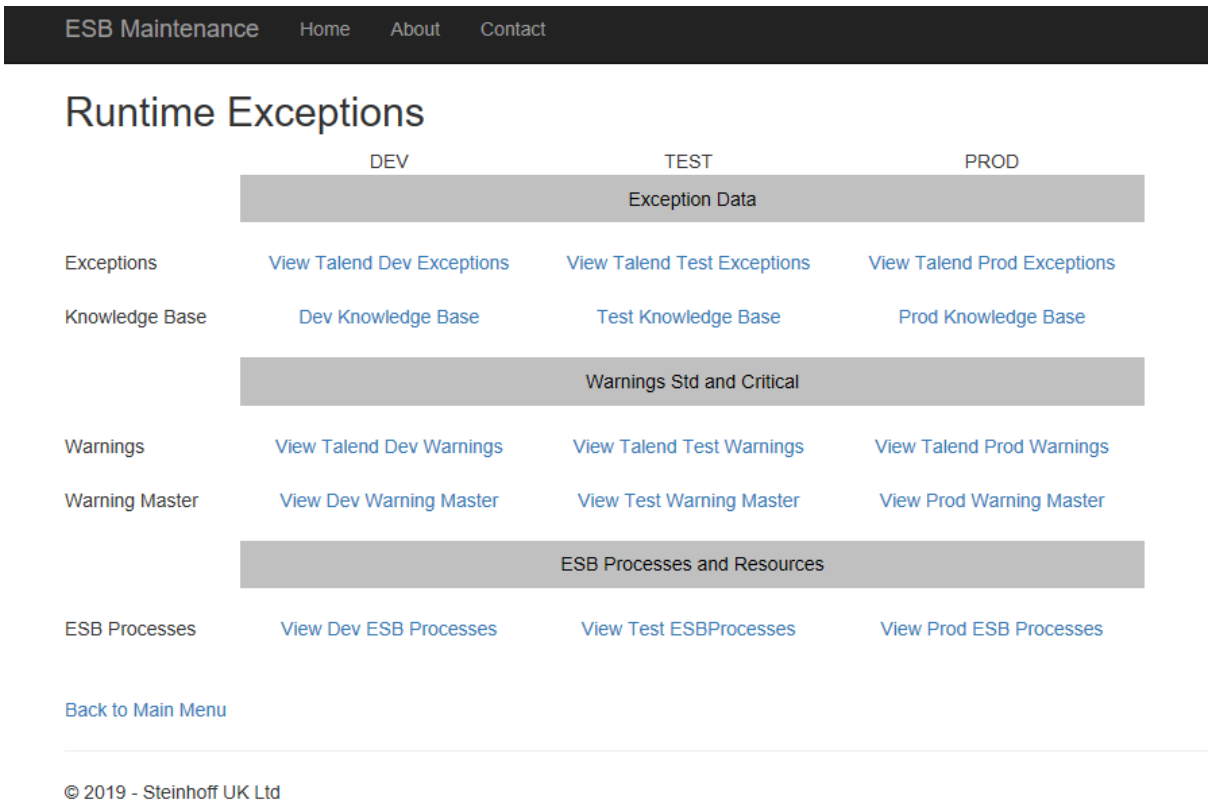
Or http://10.167.195.31/DBTableMaintenance

It is an intranet application so will require the vpn client if accessing from outside the domain. You will require a username (your e-mail address) and a password to access the system.

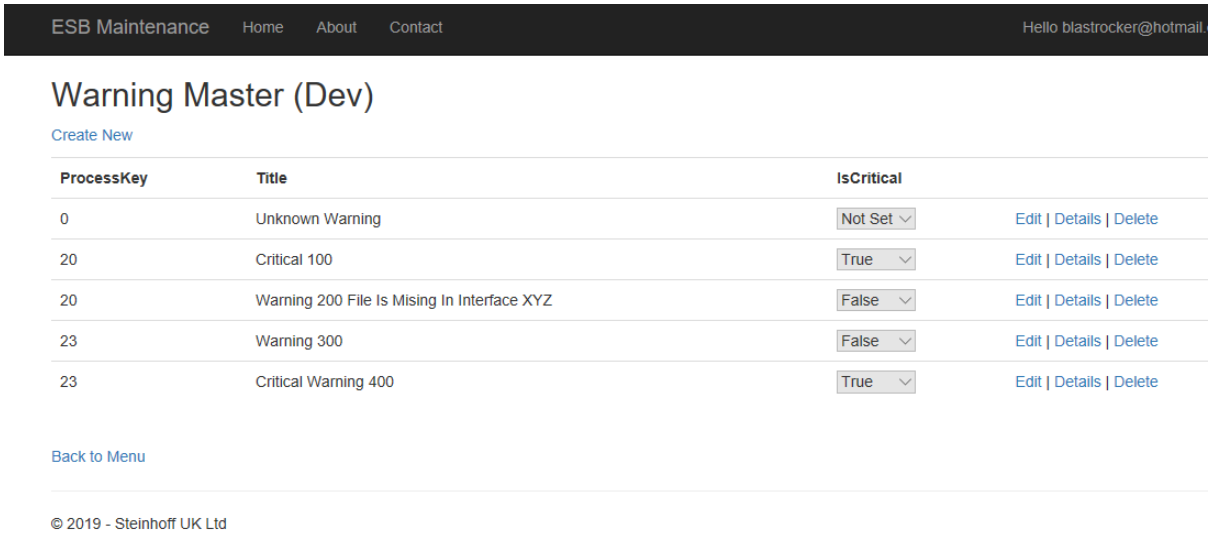
After accessing the system and logging in you will have the following main menu.



Select the exceptions option and this will take you to the exception/warning menu. You will notice this has the options for all 3 ESB environments; DEV, TEST and PROD



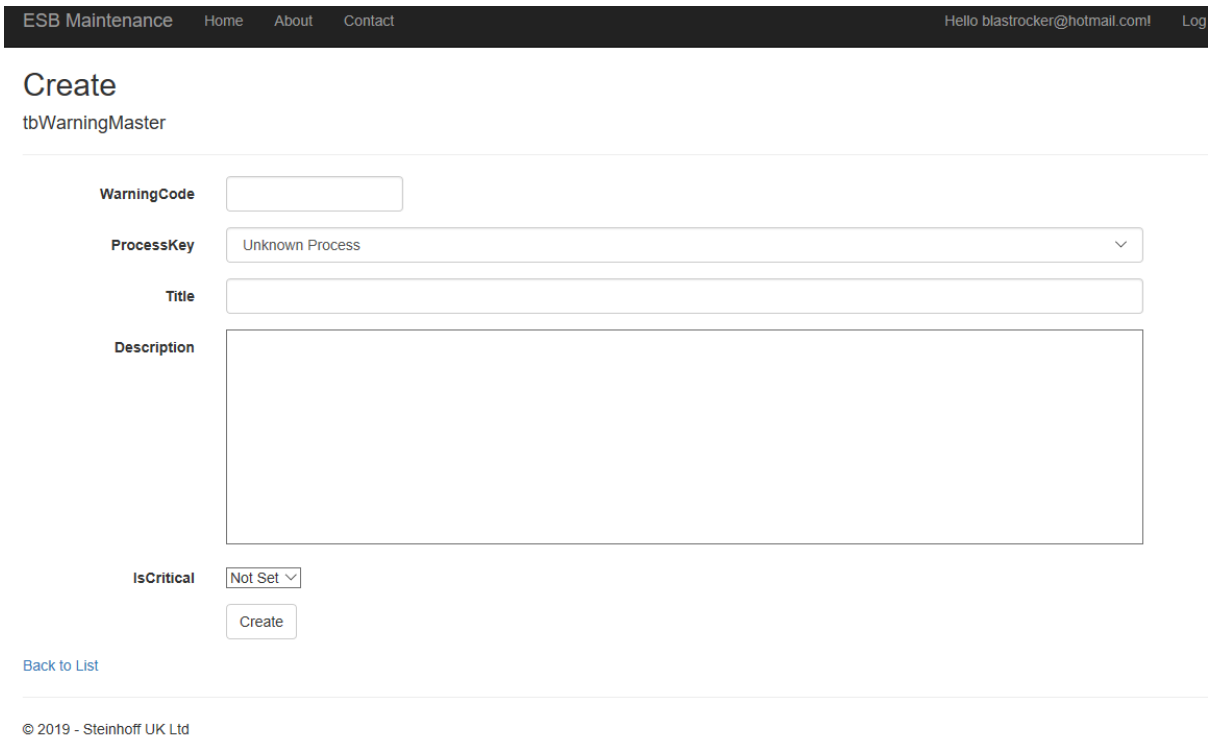
Go down to the Warning Master row and select View Dev Warning Master to get a list of the master templates defined for warnings in this environment.



At this point you can click on the Create New option to add a new template or against each current option there are options to edit, show details or delete.

1.3.1 Creating a new Warning Master

Click on Create New to show a new Master being added:



Add the Warning Code noting that this must be unique.

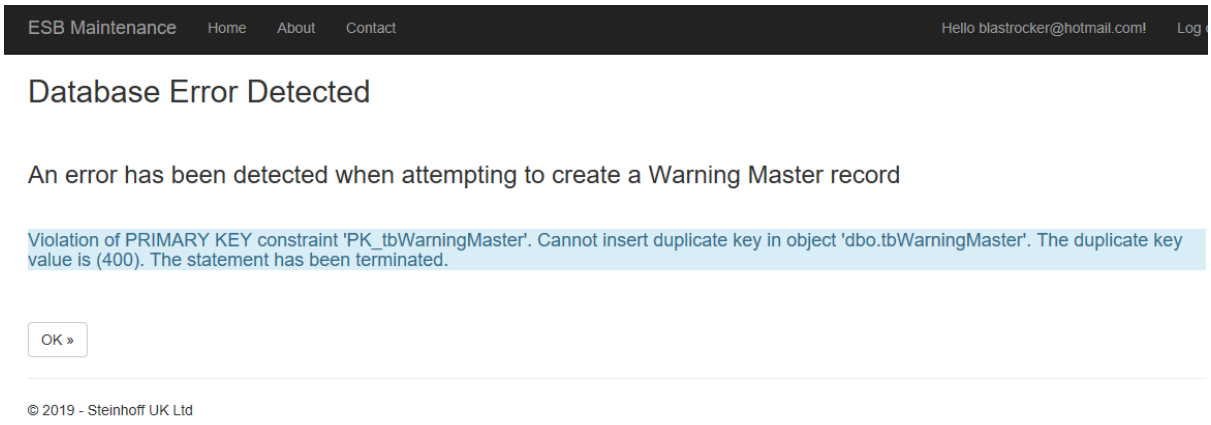
Select the Talend process from the drop-down list.

Add a title which may be the same as what is used in Talend but can be more descriptive if required

Add a description which is free test and contain as much information about the warning and how to remedy it as required.

Finally select from the option list whether it is a critical warning or not (a not set value is also treated as being a standard warning)

Provided the warning code is unique, the record will be saved as a new Warning Master. If the code is already used however you will receive an error similar to below.



The database has been set up to not allow duplicate codes and this screen is informing you of that. On dismissing the error control will return to the list of master and creation will need to commence from the start. As a rule of thumb only put minimal data in when creating the record then once it is inserted it can be modified.

1.3.2 Editing a Warning Master

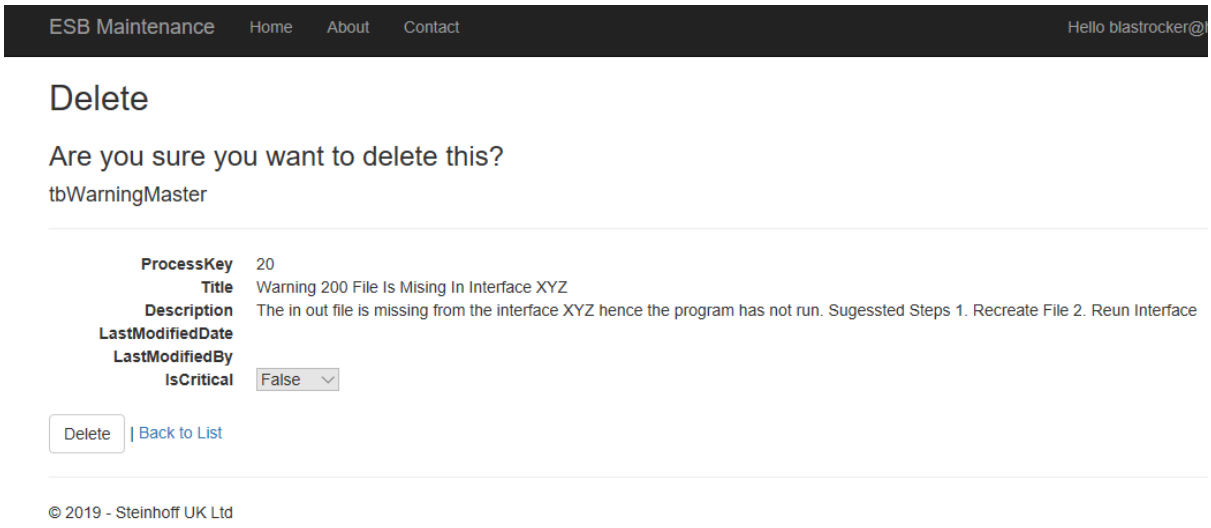
Working from the list of records scroll down to the master record of interest then click on the edit link at the right-hand end of the line. This will bring up a screen very similar to the create screen the only differences being the inability to change the warning code field and the date last modified and who by are shown as read only fields. The reason for this being that warning code is the primary key value for the record and changing of primary keys is not allowed in the database.

If you inadvertently create a record with the wrong code the only way to rectify it is to delete the record and create a new one with the correct code.

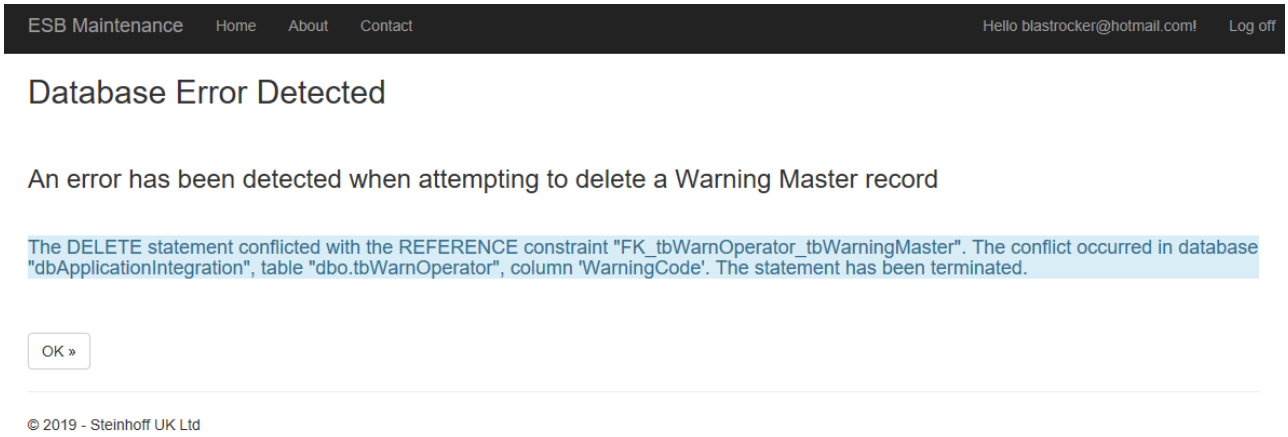
Selecting the details rather than edit option will present the same data but in a completely read only format.

1.3.3 Deleting a Warning Master

The framework allows the deletion of master record if they are not used by the system. The usual circumstance would be when they have been added with the wrong code. Clicking on the delete option will bring up a confirmation box allowing verification of the action.



Confirmation of the delete action will tell the system to attempt to delete the record from the database. Note that only unused records can be removed. If there are warning records already in the system that are linked to this master then the deletion will not be allowed. In database terminology the system is set up to enforce referential integrity.

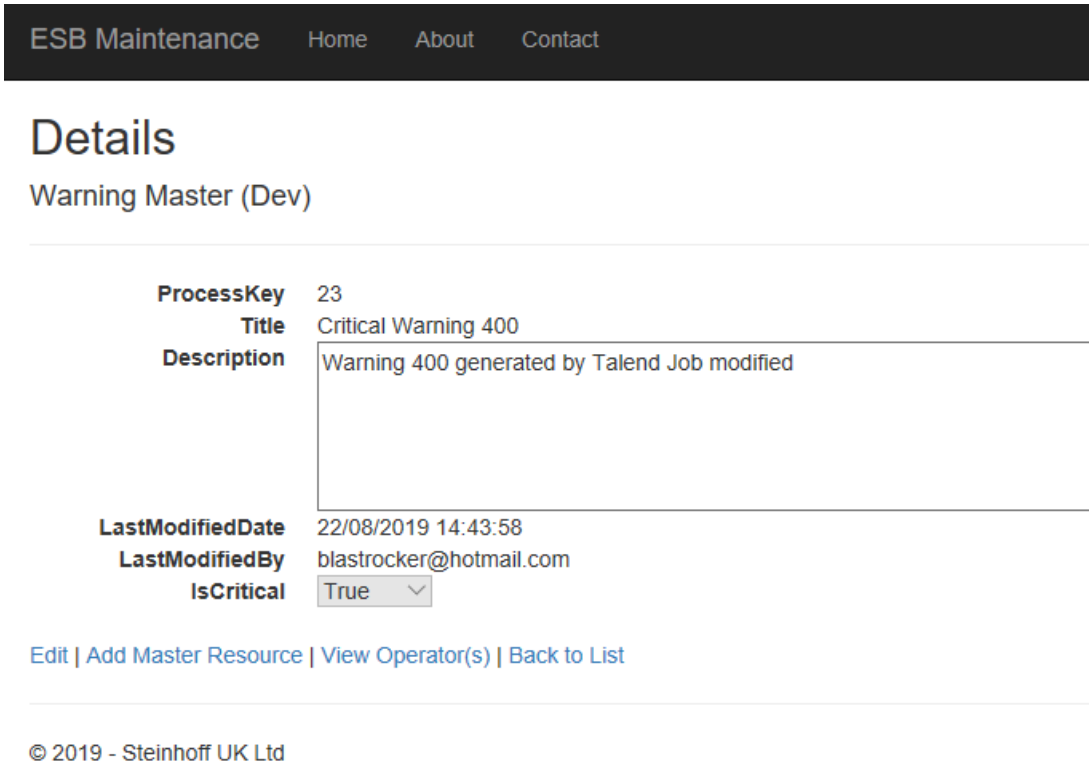


1.4 Assign operators to the master

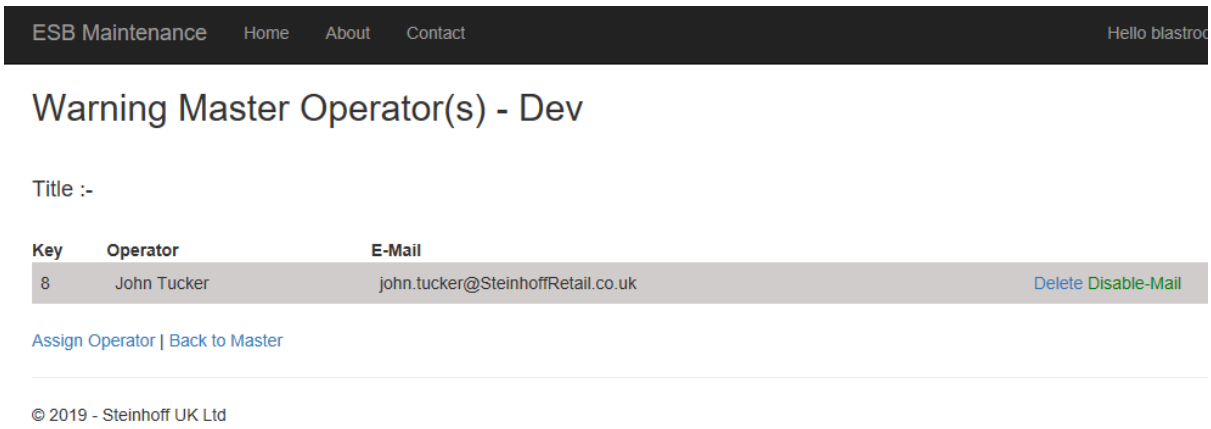
An operator is defined as a person who receives notification of the occurrence of a warning or exception by e-mail. They may be the person responsible or they could be an interested party. It is possible to assign multiple operators to a warning master and each operator will receive e-mail notification unless explicitly disabled on the set up.

Currently more relevant to exceptions rather than warnings but that could change in the future, the operator mechanism is used to automatically generate fresh service tickets from the data by setting the fresh service e-mail up as an operator.

To access the warning operator set up select an item from the list of masters and click on details. This will display a screen of information about the master

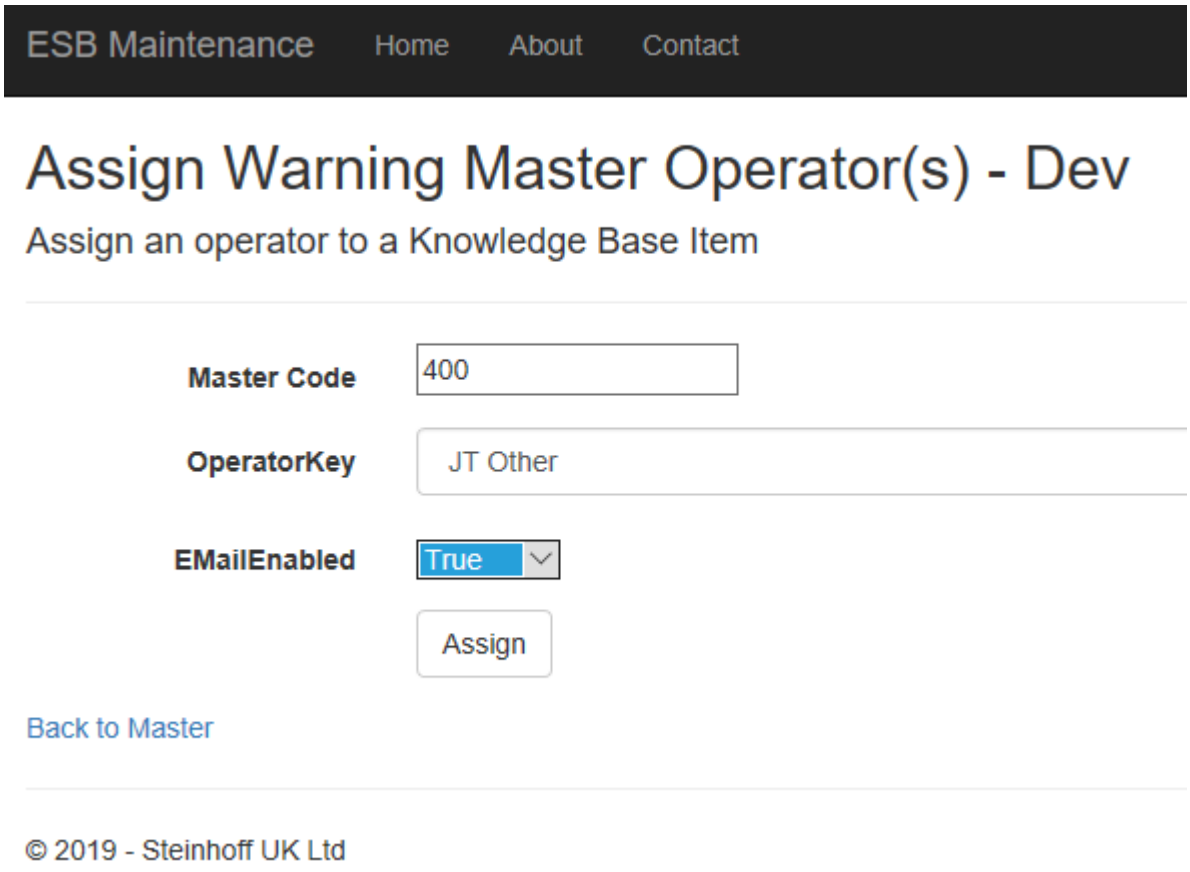


There is an option at the bottom of the page that will say either Assign Operators or View Operators depending on whether any previous operator assignment has occurred for this master record. Click on this option to call the assignment screen.



The screen lists the current operators assigned to the master. These can't be edited but they can be deleted and there is an option at the right-hand end of each row which will do this subject to the usual confirmation.

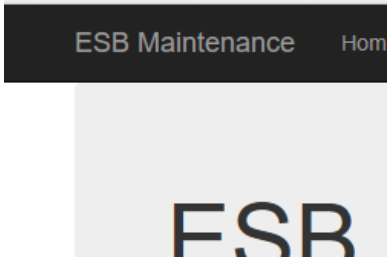
To assign further operators to this master click the assign operators option at the bottom of the screen.



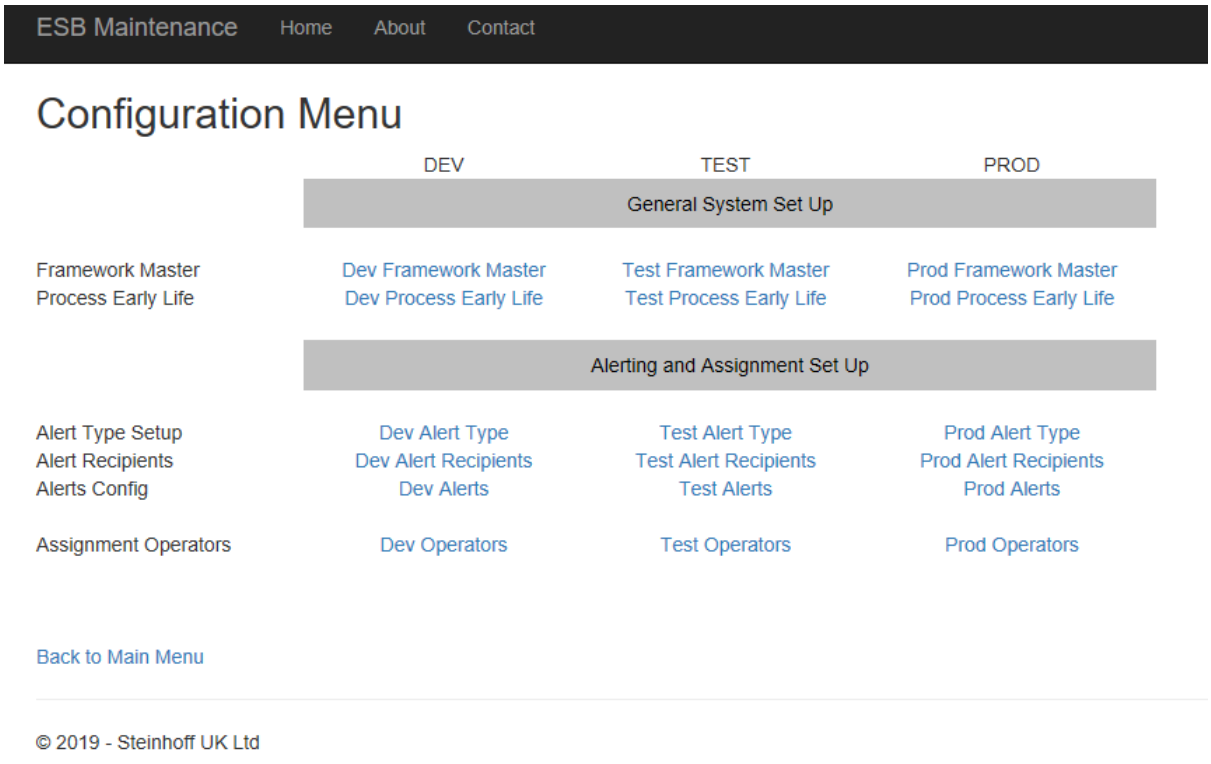
The warning code has already been entered and cannot be changed. Select the required operator from the drop-down list, confirm whether e-mail should be enabled for this operator and click on assign. The operator is now linked to the master and if you repeat the actions from the previous section they will be displayed in the list.

Only operators that have previously been recorded on the system can be assigned to the master. If the operator doesn't exist use the following instruction to create them

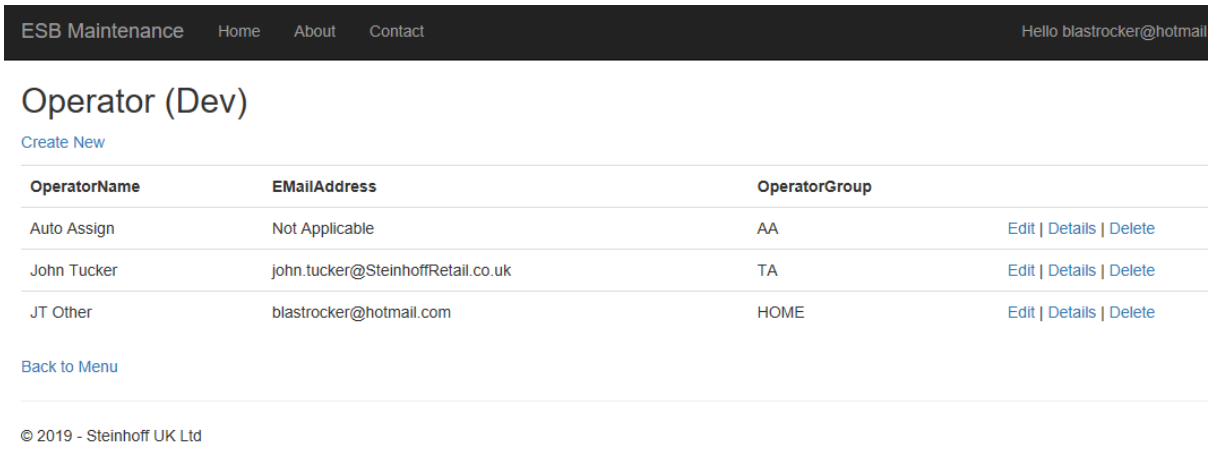
Click on the ESB Maintenance option in the top left-hand corner of the screen. This is a shortcut that will take you straight back to the main menu.



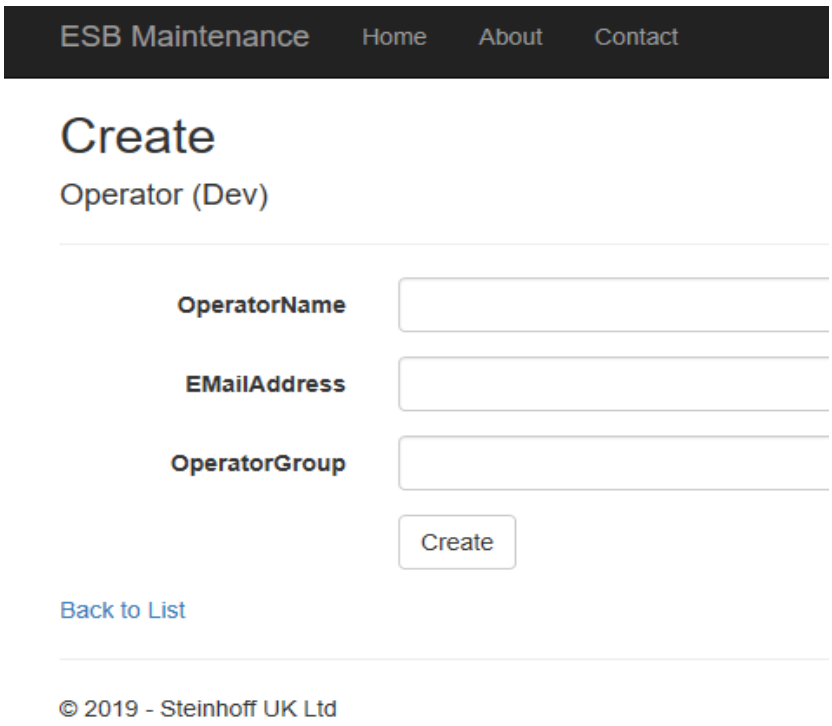
From the main menu select the Configuration option which will call the config menu



From the assignment operators row select the action appropriate to the environment you are using. In this case click on Dev Operators.



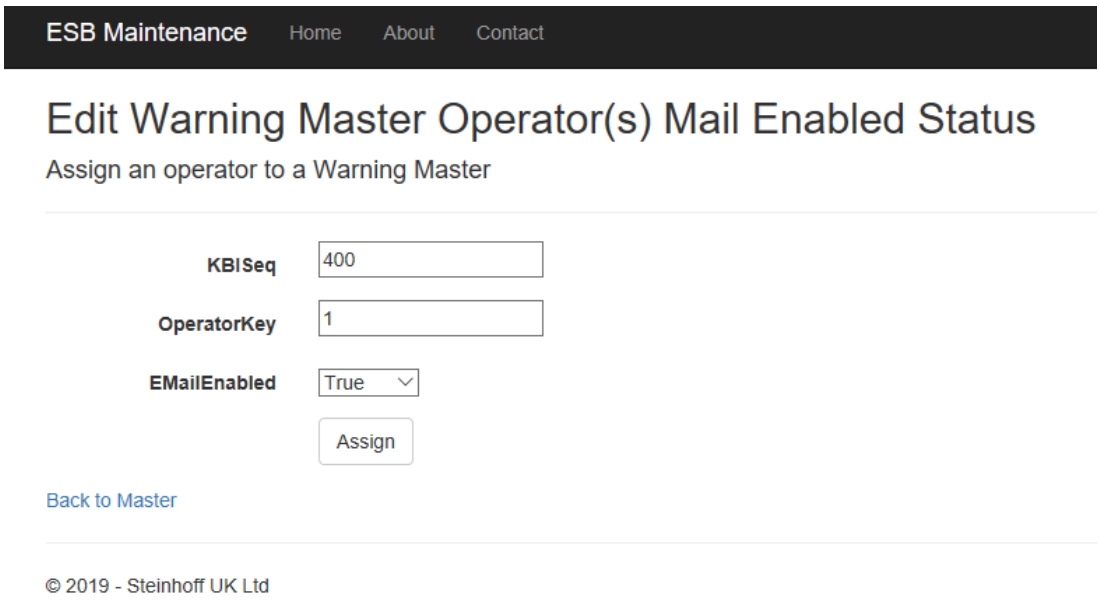
A list of the operators currently set up for the Dev system is shown. Click on Create New to add a new one. This will call the operator creation screen. Add the operator name which is a friendly label together with the e mail address. The operator group field is not currently used but may be in future versions I just use the default TA for now.



Click create and the operator will now appear in the list and can be assigned to warning masters or the exception equivalent.

Change Mail Status of Operators

In the list of operators for the warning master you will notice an option at the left-hand end of each row to enable or disable mail depending on the current status. Clicking this option will call the following screen which allows changes to be made to the e-mail status of the warning operator. This is the only change allowed once an operator has been assigned.



When a notification of a warning is sent the first part of the e-mail includes a list of all operators assigned to the warning master. This is useful when being sent to fresh service to show who the automatically raised ticket should be assigned to. Normally the system would also send notifications

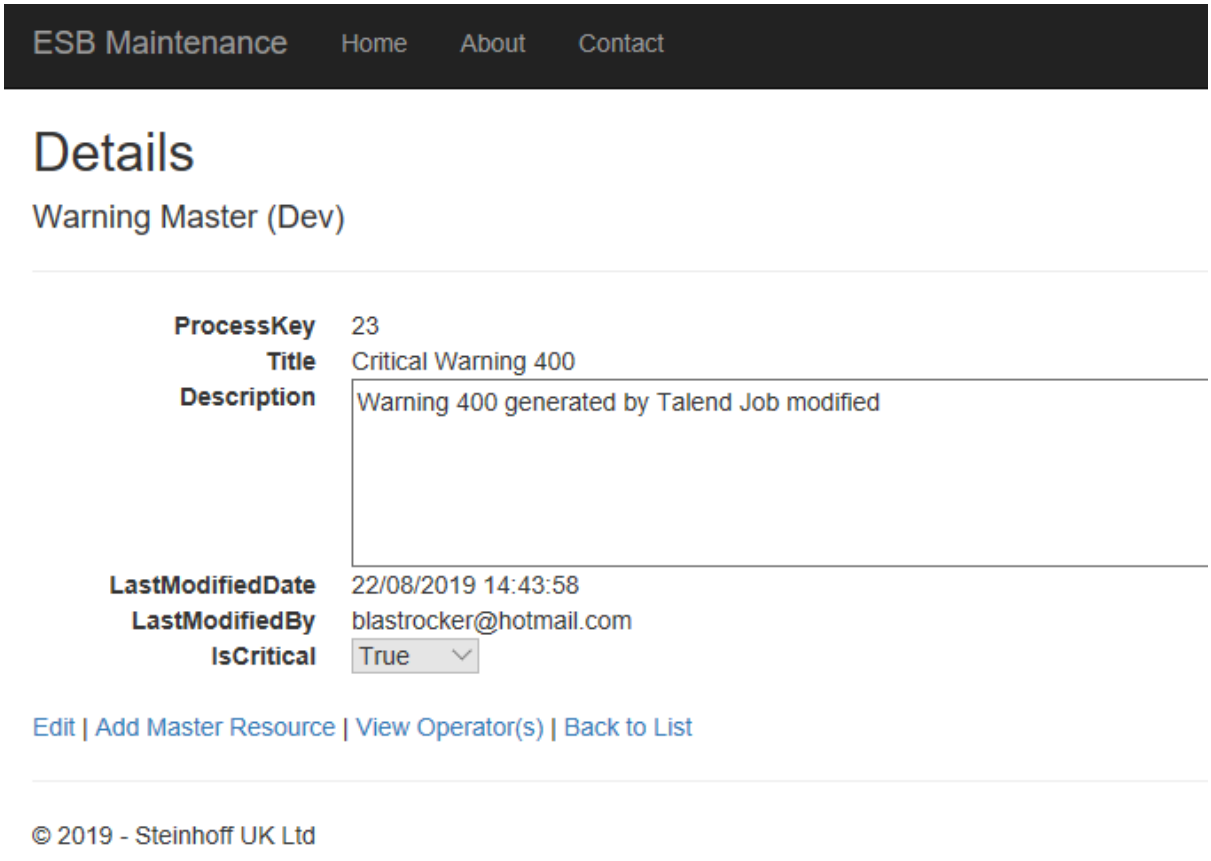
to all the operators specified however this may not be desired as the problem should be managed only through fresh service. Disabling the e-mail prevents these direct notifications being sent but still keeps the operator in the list of assignees. Any combination of notifications can be modelled through this process.

This may be more prevalent in exceptions rather than warnings at this stage but the facility is built in anticipating future changes.

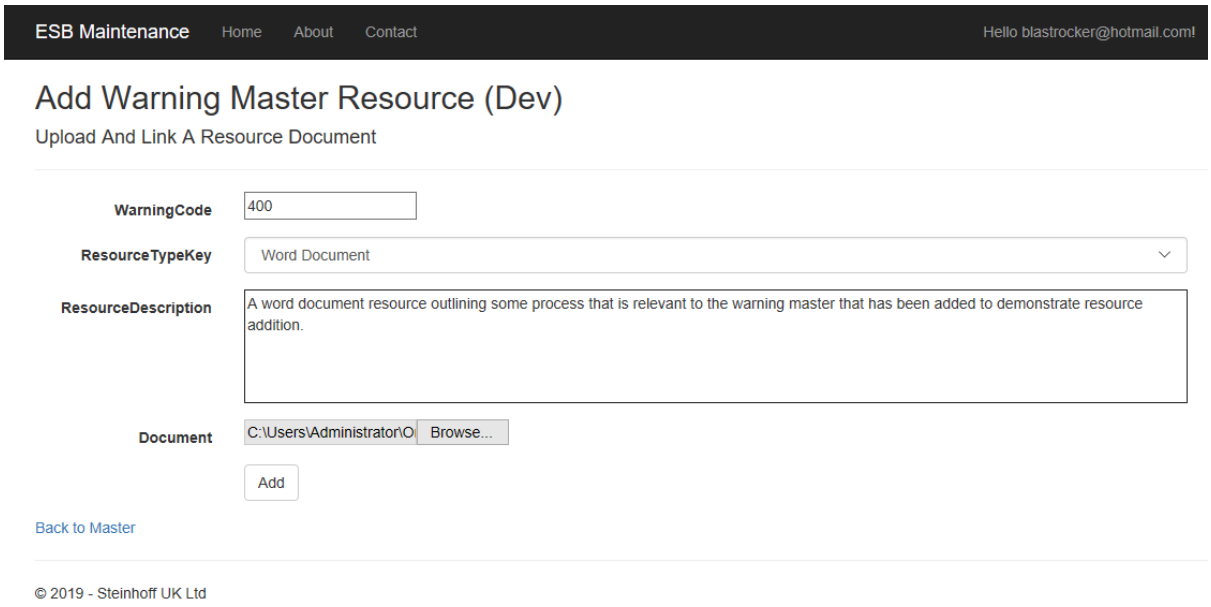
1.5 Optionally add resources to the master

A resource is an external document that can be used to add additional background information to the circumstances surrounding a warning or exception. Typical examples may be a concept diagram or a specification. The documents can be uploaded and automatically associated with the master. Subsequent viewing of the master record allows the resources to be downloaded and viewed.

To add a resource to a warning master first select an item from the list of warnings and click on details which will bring up the details screen.

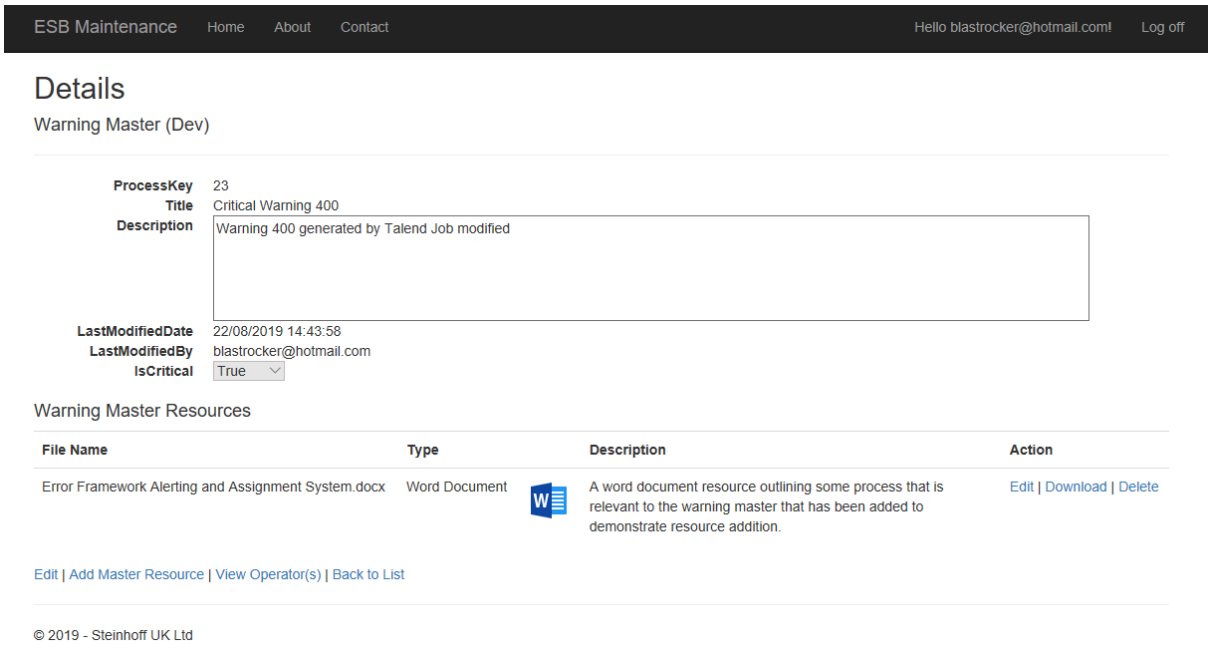


From the options at the bottom left-hand corner select Add Master Resource. The add master resource screen will be called with the warning code already entered. From the drop-down list select the type of document the resource is. Note this isn't verified so if you select the wrong type of document the resource will still be uploaded regardless. It's not critical data just a visual aid to easily see what resources are available.



Having selected the type a suitable description should be entered outlining what the resource provides and finally the document itself is selected by clicking the browse button and moving around the folder structure to find the required document.

Click on Add when complete and control returns to the warning master detail screen, the added resource can now be seen.

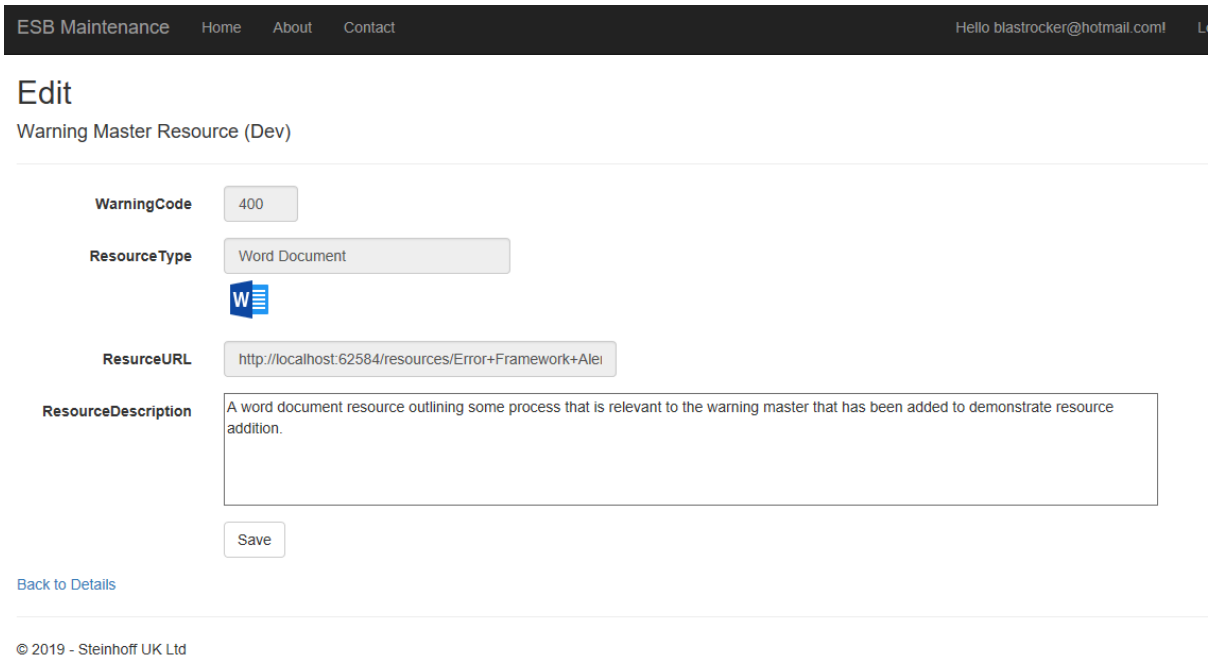


At the right-hand end of each resource line there are options to edit, download or delete the resource.

1.5.1 Editing a Warning Master Resource

Once a resource has been added the only modification allowed is to the description. If you wish to change the document type or the document itself the resource must be deleted and recreated.

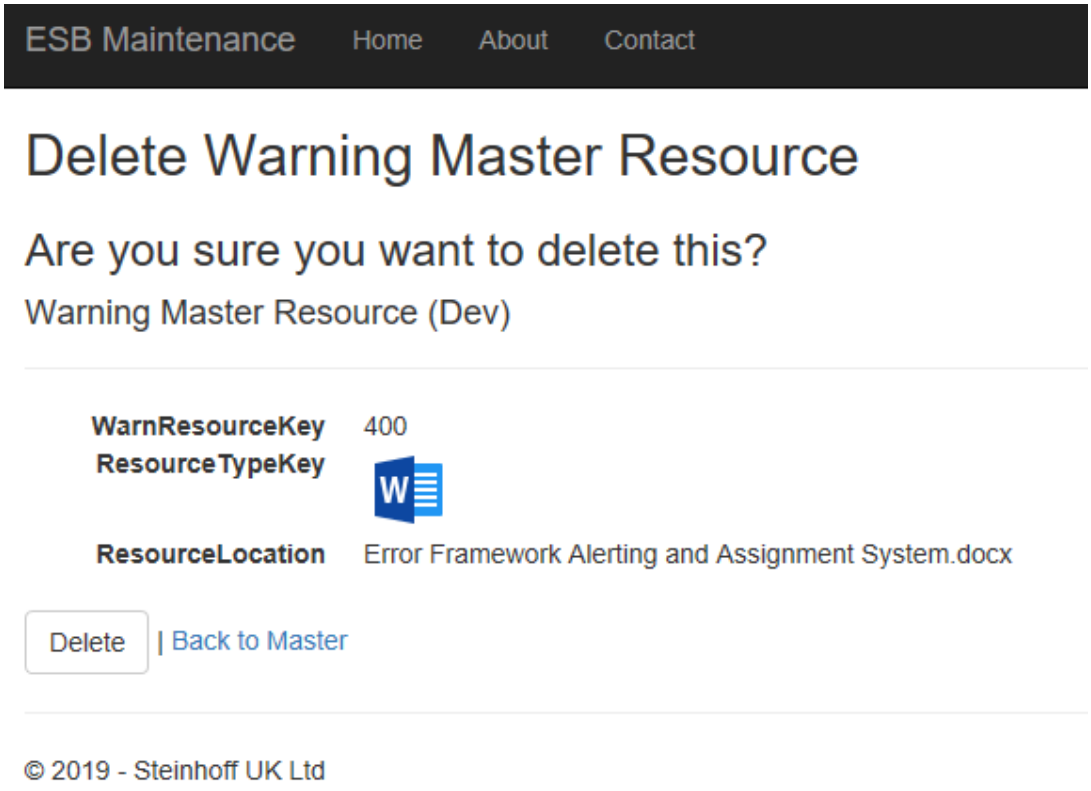
Click on edit to call the resource editor.



Modify the description as required and click the save button and control returns to the detail screen where the modified description will be visible.

1.5.2 Deleting A Warning Master Resource

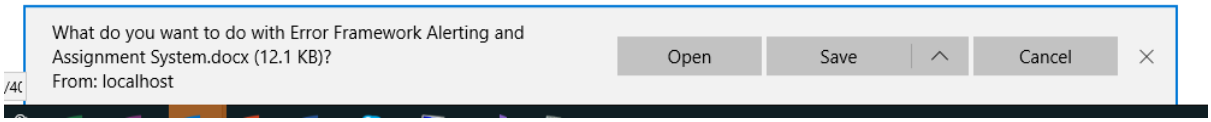
Selecting the delete option on resource line will ask for confirmation of the intention to delete as below:



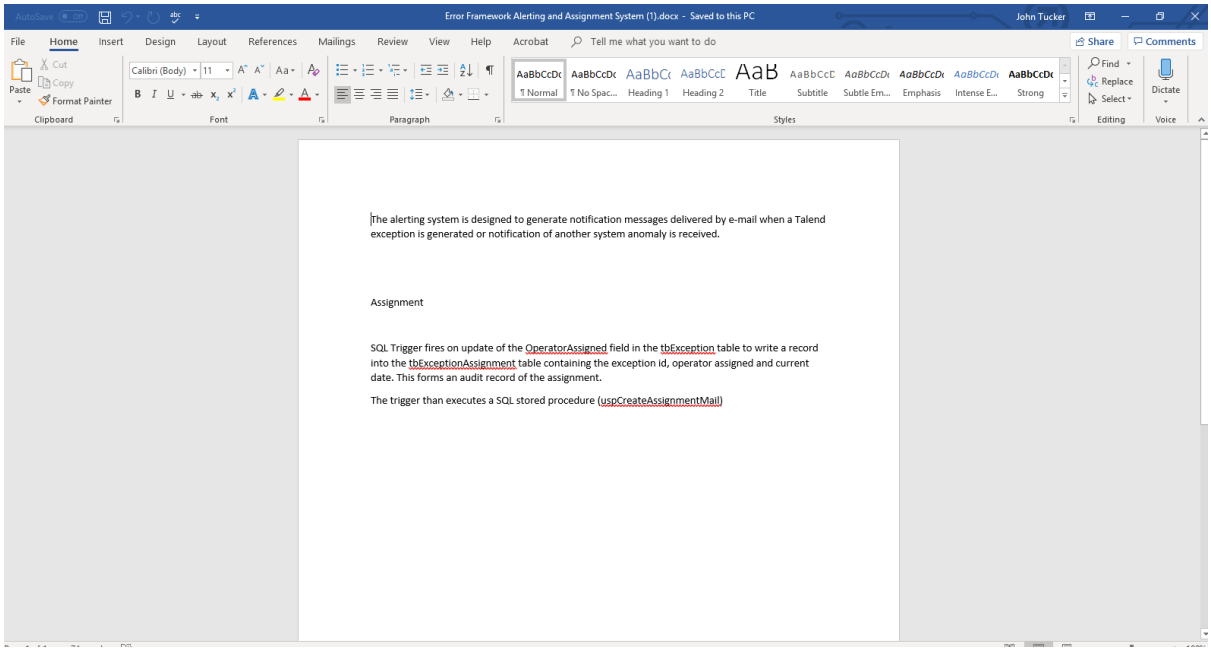
Selecting delete will remove the resource or Back to Master will cancel the operation. In either case control will return to the details screen where if the deletion was confirmed, the resource will no longer be shown. Note that deleting a resource removes the reference to the file in the database but doesn't physically remove the file which will still be present in the central file system. Periodic maintenance will manually clean down unused resource files.

1.5.3 Download A Resource

Selecting the download option for a resource will cause the following dialog box to appear at the bottom of the screen.



Selecting open will display a copy of the document in it's native application. In this case it was a word file so it opened in MS Word.



Clicking save will save a copy of the file to your local download folder or alternatively you can select save as and place the copy in a location of your own choice. These are all standard mechanism that have been hooked into the application.

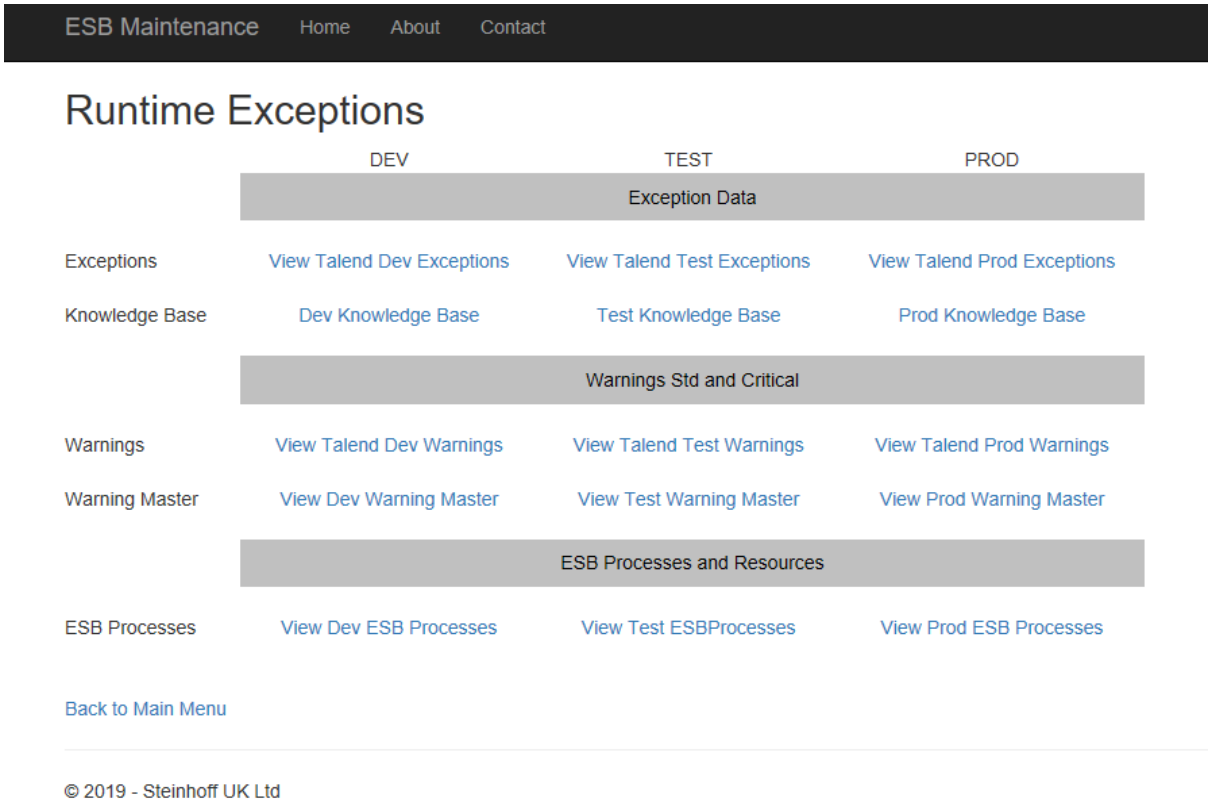
1.6 Process Resources

In addition to saving resource against a Warning Master it is also possible to have resources linked to a process. These would include process descriptions/specifications and other items that apply to all warnings for that process. Rather than repeating these it is possible to have a single copy that can be implicitly referenced by all warnings or exception relevant to the process. A separate section exists for process resource so please refer to it for further details.

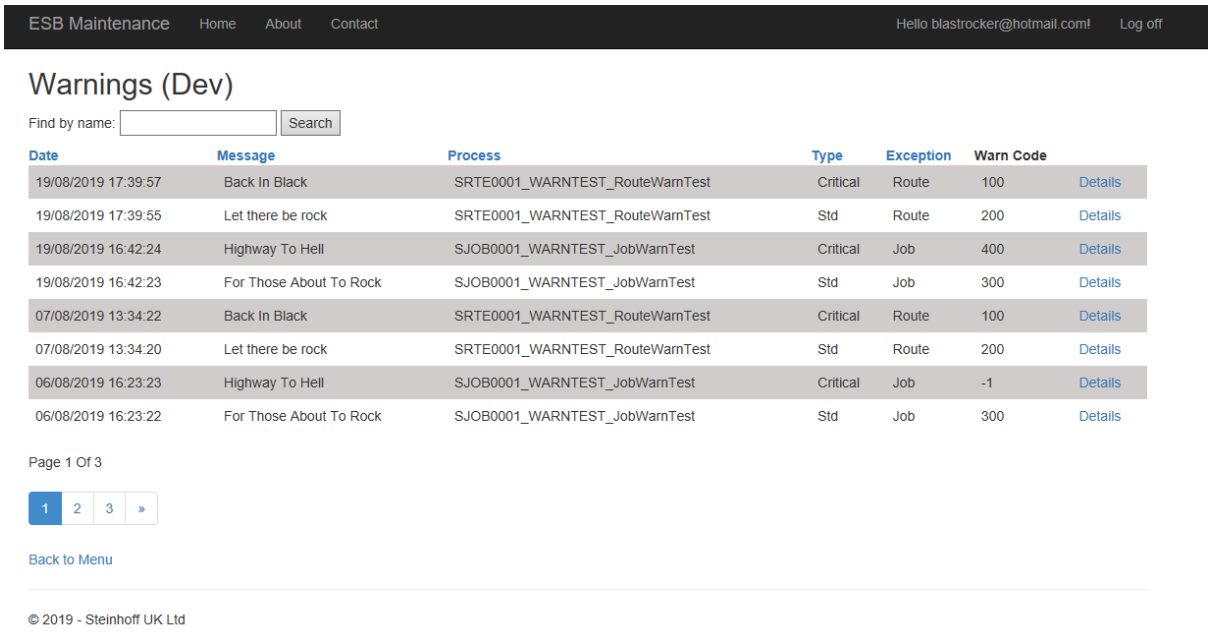
Any process resources will automatically appear in the warning master details. They can be downloaded here but not edited or deleted.

2. Warning Assignment

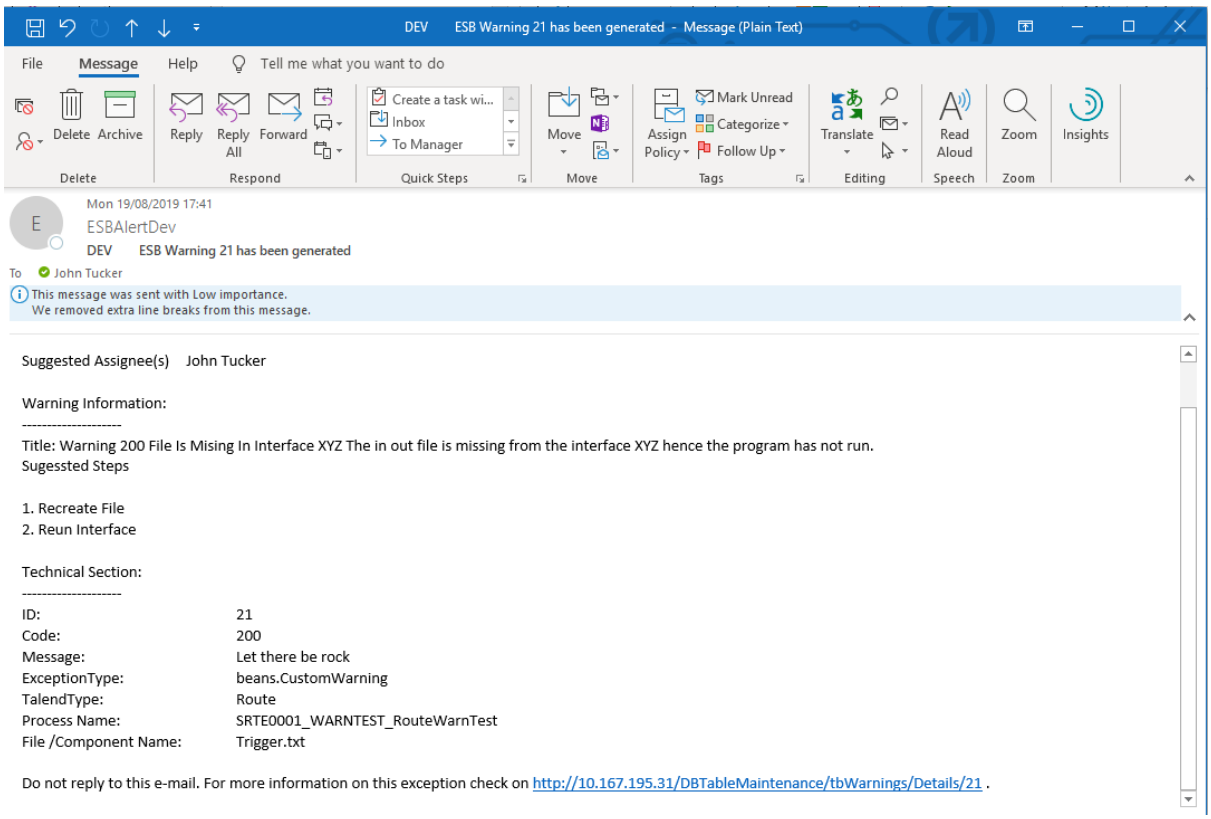
With the warning master correctly set up the process of assigning warnings generated by Talend to the defined operators can begin. The warnings are initially loaded into a warning table for each environment and these can be accessed via the View Talend Dev Warnings option of the runtime exceptions menu.



This will present a paged list of the captured warning data.

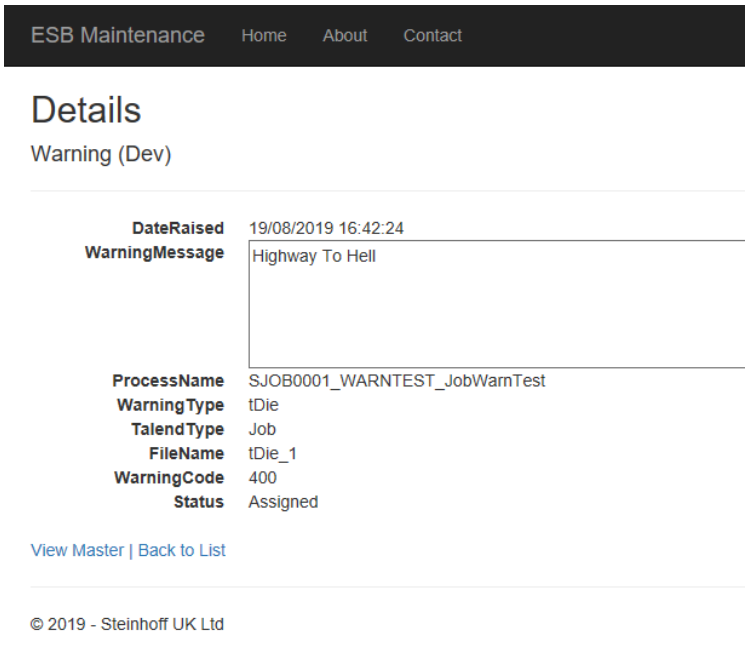


After a warning has been loaded into the system a routine which runs every minute picks up the warning data and links it with the master record for the warning code. This data is then used to generate the body for an e-mail which is then sent to the operators assigned to the warning master.



The warning information at the top of the e-mail is extracted from the master record and the technical section contains the data from the warning itself. Recipients of the e-mail can click on the link which will take them to the warning record within the system from which all resource data can be accessed.

Accessing Master Information from a Warning



When selecting the warning details record there is an option at the bottom of the screen to view master. Clicking on this take you to the warning master record as per this document but without having to come out and go through the menu. The only subtle differences are in the navigation which will return you to the warning record rather than the list of masters.