



JazzConnect-EMAIL-Reader Edition

Installation & User Guide Version 5.4.0

All in this document is owned by Kintrax Limited. Any person or organisation is hereby not authorised to copy, print, or distribute this document or any part of, without the express written agreement of Kintrax Limited.

Note that any product, process or technology described in the document may be the subject of other Intellectual Property rights reserved by Kintrax Limited and are not licensed hereunder.

TRADEMARKS

Kintrax Limited and its logos are registered trademarks of Kintrax Limited. All other product names mentioned herein are the trademarks of their respective owners.

THIS PUBLICATION IS PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT. THIS PUBLICATION COULD INCLUDE TECHNICAL INACCURACIES OR TYPOGRAPHICAL ERRORS. CHANGES ARE PERIODICALLY ADDED TO THE INFORMATION HEREIN; THESE CHANGES WILL BE INCORPORATED IN NEW EDITIONS OF THE PUBLICATION. KINTRAX LIMITED MAY MAKE IMPROVEMENTS AND/OR CHANGES IN THE PRODUCT(S) AND/OR THE PROGRAMME(S) DESCRIBED IN THIS PUBLICATION AT ANY TIME.

- ◆ CM-InPractice™, CM-InSync™, CM-InView™, CM-InTouch™, CM-CQDocumenter™, JazzConnect™, JIRAConnect™ are trademarks of Kintrax Ltd.
- ◆ ClearCase®, Team Concert®, Jazz are registered trademarks of IBM Rational Software Corporation.
- ◆ UNIX™ is a trademark of AT&T Bell Laboratories.
- ◆ Sun, SunOS, Solaris, NFS, PC-NFS, are trademarks or registered trademarks of Sun Microsystems, Inc.
- ◆ Windows NT™ and Visual C++® are trademarks of Microsoft Corporation.
- ◆ Linux is written and distributed under the GNU General Public License.
- ◆ HP-UX® is a registered trademark of Hewlett-Packard Company.
- ◆ AIX® is a registered trademark of IBM Corporation.

Table of Contents

1	INTRODUCTION	6
2	PRODUCT OVERVIEW	7
2.1	KEY COMPONENTS	7
2.2	SYSTEM ARCHITECTURE	7
3	PRE-REQUISITES	8
3.1	AT THE JAZZ SITE	8
3.2	AT THE INSTALL SITE	8
3.3	EMAIL CLIENT	8
4	PREPARING TO INSTALL	8
4.1	LICENSE FILE.....	8
5	INSTALLING THE JAZZCONNECT-EMAIL READER TOOLSET	9
6	CONFIGURING JAZZCONNECT-EMAIL READER EDITION	10
6.1	THE EMAIL READER CONFIGURATION FILE.....	10
6.1.1	<i>Configuration File Hierarchy.....</i>	<i>10</i>
6.1.2	<i>Configuring the Parameters.....</i>	<i>11</i>
6.2	RUNNING THE EMAIL READER APPLICATION.....	11
6.2.1	<i>Manual (via Command Line).....</i>	<i>11</i>
6.2.2	<i>Automatic (as a Schedule).....</i>	<i>12</i>
6.2.3	<i>Automatic (as a Service).....</i>	<i>12</i>
7	CONNECTOR TESTING	13
7.1	SEND AN EMAIL TO THE SYNCHRONISED INBOX.....	13
7.2	RUN THE EMAILREADER.BAT FILE	13
7.3	CHECK WORK ITEM IN RATIONAL TEAM CONCERT	13
8	USING THE EMAIL READER	15
8.1	CREATING A WORK ITEM BY EMAIL	15
8.2	UPDATING AN EXISTING WORK ITEM BY EMAIL	16
8.3	EMAIL USAGE EXAMPLE 1	16
8.4	EMAIL USAGE EXAMPLE 2	17
8.5	EMAIL USAGE EXAMPLE 3	18
8.6	UPDATING THE STATUS OF A WORK ITEM BY EMAIL	18
8.7	EMAIL USAGE EXAMPLE 4	19
8.8	SETTING UP REVIEW/APPROVAL BY EMAIL.....	20
8.9	APPROVING/REJECTING A WORK ITEM BY EMAIL.....	21
8.10	EMAIL USAGE EXAMPLE 5	21
8.11	EMAIL USAGE EXAMPLE 6	21
9	APPENDIX A – INSTALLATION REQUIREMENTS TABLE.....	23
10	APPENDIX B – PARAMETERS	24
10.1	ATTRIBUTES SET BY THE SYSTEM.....	29
10.2	BODY-FIELDS.....	30
11	APPENDIX C – COMMAND LINE DISPATCHER	32
11.1	ENCODE PASSWORD.....	32
11.2	CREATE USER.....	32
11.3	DUMP FIELDS	33
11.4	PROCESS EMAILS	33
11.5	TEST LICENSE.....	34

11.6	LDAP TEST	35
11.7	HOST DATA.....	35
11.8	SMTP TEST.....	35
12	APPENDIX D – LDAP CONFIGURATION	36
13	APPENDIX E - SMTP CONFIGURATION	37

Figures

Figure 1: System Architecture.....	7
Figure 2: Command Line Prompt running the EmailReader.bat file	11
Figure 3: Test send email	13
Figure 4: Work Item successfully created in RTC.....	14
Figure 5: Setting attribute values via email	15

1 Introduction

This document describes the installation of JazzConnect-Email Reader Edition, a tool which enhances the use of Rational Team Concert by automatically creating work items in RTC from emails sent to a specific inbox or set of inboxes.

The document also includes the initial configuration of parameters required to enable the tool to function in your environment.

2 Product Overview

JazzConnect-Email Reader Edition is an application that provides middle-ware support to the automation of Work Item creation in Jazz using the Work Item Management tool in the Jazz architecture's 'Change and Configuration Management' application (RTC).

2.1 Key components

JazzConnect-Email Reader Edition is stand-alone software. All configuration and customization can be done using the files located in its own installation directory.

The Email Reader is a java application consisting of our java libraries, Jazz's JPJC libraries, configuration files and executable files.

The Email Reader stores these files and then uses an internal scheduling process to determine any creation/update requirements.

2.2 System Architecture

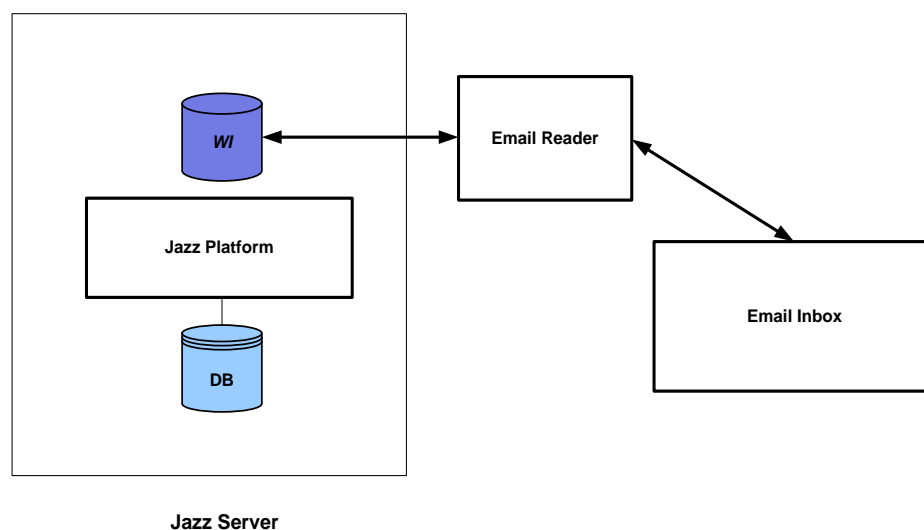


Figure 1: System Architecture

3 Pre-requisites

JazzConnect-Email Reader Edition **requires** the following additional applications in order to correctly function.

3.1 At the Jazz Site

- Rational Team Concert Version 4.0 or later up to version 6.0.x

3.2 At the Install Site

- Jazz Plain Java Client downloaded from jazz.net (See point 4 of Installing the JazzConnect-Email Reader toolset)

3.3 Email Client

- The required credentials for accessing the designated inbox(es) to be synchronised - where emails will be converted into work items in Rational Team Concert. (It is best to create a new email account so there's no interference with existing mail in the inbox(es).

4 Preparing to install

More details on the installation/configuration process can be found in “

Appendix A – Installation Requirements Table” on page 23.

4.1 License File

Each JazzConnect-Email Reader Edition installation requires a license file. The license file will be supplied by Kintrax and must be stored in a browse able directory to be accessed during installation.

5 Installing the JazzConnect-Email Reader toolset

1. Decide on the JazzConnect-Email Reader Edition installation area. i.e. C:\Program Files\KINTRAX. The installation location does not need to be part of the Jazz server installation or on the same machine.
2. Copy the `cml_jazz-emailreader.tar.gz` file to the installation location and unzip it.
3. Copy the license file (supplied by Kintrax) to this directory. (i.e. C:\Program Files\KINTRAX\jazz-emailreader\conf) making sure the file is called `license.lic`
4. Navigate to <http://jazz.net> and download the 'Plain Java Client Libraries' that matches your version of Jazz Team Server. (It can be found in the 'All Downloads' tab.)
i.e.
 - a. 4.0 - <https://jazz.net/downloads/rational-team-concert/releases/4.0?p=allDownloads>
 - b. 4.0.1 - <https://jazz.net/downloads/rational-team-concert/releases/4.0.1?p=allDownloads>
 - c. 4.0.2 - <https://jazz.net/downloads/rational-team-concert/releases/4.0.2?p=allDownloads>
 - d. 4.0.3 - <https://jazz.net/downloads/rational-team-concert/releases/4.0.3?p=allDownloads>
 - e. 4.0.4 - <https://jazz.net/downloads/rational-team-concert/releases/4.0.4?p=allDownloads>
 - f. 4.0.5 - <https://jazz.net/downloads/rational-team-concert/releases/4.0.5?p=allDownloads>
 - g. 4.0.6 - <https://jazz.net/downloads/rational-team-concert/releases/4.0.6?p=allDownloads>
 - h. 5.0 - <https://jazz.net/downloads/rational-team-concert/releases/5.0?p=allDownloads>
 - i. 5.0.1 - <https://jazz.net/downloads/rational-team-concert/releases/5.0.1?p=allDownloads>
 - j. 5.0.2 - <https://jazz.net/downloads/rational-team-concert/releases/5.0.2?p=allDownloads>
 - k. 6.0.0 - <https://jazz.net/downloads/rational-team-concert/releases/6.0?p=allDownloads>

These libraries enable the Email Reader to communicate with the Jazz server(s) so the version downloaded here must match the server version.

5. Unzip and copy all of the '.jar' files from the 'Plain Java Client Libraries' download into the '<Email-Reader-Installation>\jppc-lib'.

6 Configuring JazzConnect-Email Reader Edition

Configuration of the JazzConnect-Email Reader Edition is achieved in two steps. Firstly, we configure an xml file and secondly, we configure the command line for the EmailReader.bat file that carries out the actual process.

Note: For the purposes of this configuration example, we are using the Joe Blogg's Project in Jazz.

Note: Information from the Pre-requisites section on page 8 will be required to complete this task.

6.1 The Email Reader Configuration File

Configuring which email inboxes are processed by the Email Reader is achieved via an .xml configuration file. An example file Demo-Config-For-Install.xml can be found inside /conf folder in the installation package.

6.1.1 Configuration File Hierarchy

The xml file is arranged hierarchically as follows:

- **Global** – this section holds default settings for all inboxes handled by this Email Reader instance. Values specified here, unless also specified in one of the 'Inbox' or 'User' sections, will be used.
- **Inbox** – this section contains settings for an inbox. Values specified here, unless also specified in one of the 'User' sections, will be used. If a parameter is specified in the 'Global' section and in an 'Inbox' section, the value from the 'Inbox' section will be used. The parameters set here will apply to all emails in the specified inbox.
- **User** – this section contains settings for an email address or set of email addresses (which can be specified by a regular expression). Values specified here will be used. If a parameter is also specified in the 'Global' section or in an 'Inbox' section, the value from the 'User' section will be used. The parameters set here will apply to all emails in the specified inbox sent by the specified email address or set of email addresses.

Default settings are set in the 'Global' section so that if a parameter is not specified inside an inbox specification, this default is used. As mentioned above, if a parameter is specified both in the 'Global' section and in the 'Inbox' section, the 'Global' setting is overridden by the 'Inbox' setting. The same principle applies for the inbox/user relationship.

A commented example is included in our installation package. Open the Demo-Config-For-Install.xml file in a text editor of your choosing. This file can be found in the <jazz-emailreader-install-loc>/conf folder.

Please copy/rename this demo configuration file to emailreader.xml and update accordingly.

6.1.2 Configuring the Parameters

A comprehensive list of parameters that can be specified in this configuration file are shown in Appendix B – Parameters on page 24. Many Work Item attributes can also be configured in this file and are included in the appendix also.

Note: All parameters must be surrounded by parentheses.

Note: Multiple inboxes can be configured one after the other in the emailreader.xml file.

6.2 Running the Email Reader Application

The Email Reader application can be run manually via the command line or it can be set up to run on a schedule. The option we would recommend is running it as a service. All three options are detailed below.

6.2.1 Manual (via Command Line)

1. Open a command line prompt in <jazz-emailreader-install-loc>\bin and run EmailReader.bat. (Note: For Unix, simply replace EmailReader.bat with EmailReader.sh.)
2. Type “EmailReader.bat processemails -d -p -o” and press Enter.

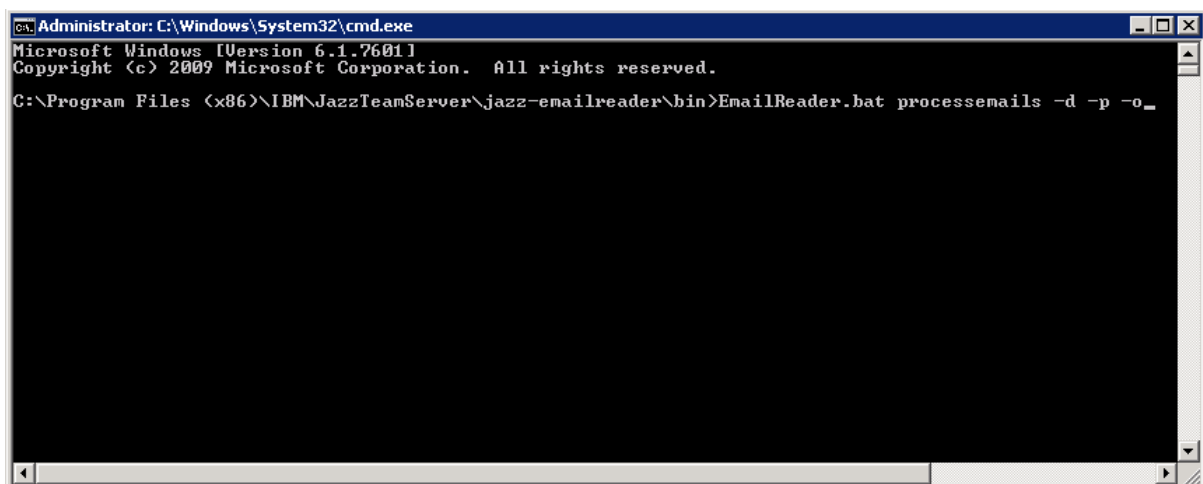


Figure 2: Command Line Prompt running the EmailReader.bat file

Note: For all of the available command line options type `EmailReader.bat --help`

6.2.2 Automatic (as a Schedule)

The command line prompt required to run the Email Reader application detailed above in section 6.2.1 can be run automatically by setting up a schedule for it.

On Windows Server 2003/2008, select Start -> Administrative Tools -> Task Scheduler. Select Create Task, set it to run every minute (or less) for an indefinite amount of time and in the 'Actions' tab, specify the command line you have decided upon.

Note: Alternatively, you could write the command line in a batch file and run that.

6.2.3 Automatic (as a Service)

In order to install the JazzConnect-Email Reader inbox processor as a daemon/service, please do the following:

1. Download and install YAJSW (Yet Another Java Service Wrapper) from <http://sourceforge.net/projects/yajsw/>
2. Copy the contents of the `jazz-emailreader\yajsw` folder to the `<path-to-yajsw>\bat` directory.
3. Update the wrapper config file in `jazz-emailreader\conf\jcemailreader.wrapper.conf` (Search for "JazzConnect-EmailReader" to determine which fields have to be updated)
4. Copy this file to the `<path-to-yajsw>\conf` directory.
5. In order to test the configuration, execute the `<path-to-yajsw>\bat\runConsoleJcEmailReader.bat` file.
6. Make any required configuration changes then install the service by executing the `<path-to-yajsw>\bat\installJcEmailReaderService.bat` file.
7. Start the service by executing the `<path-to-yajsw>\bat\startJcEmailReaderService.bat` file.

Note: The same steps can be following to install the processor as a daemon on Unix (simply replace `.bat` with `.sh` and place them in the `/bin` folder rather than the `/bat` folder).

7 Connector Testing

This section details a walkthrough of a simple test that can be carried out to verify if the connector is working correctly.

7.1 Send an email to the synchronised inbox

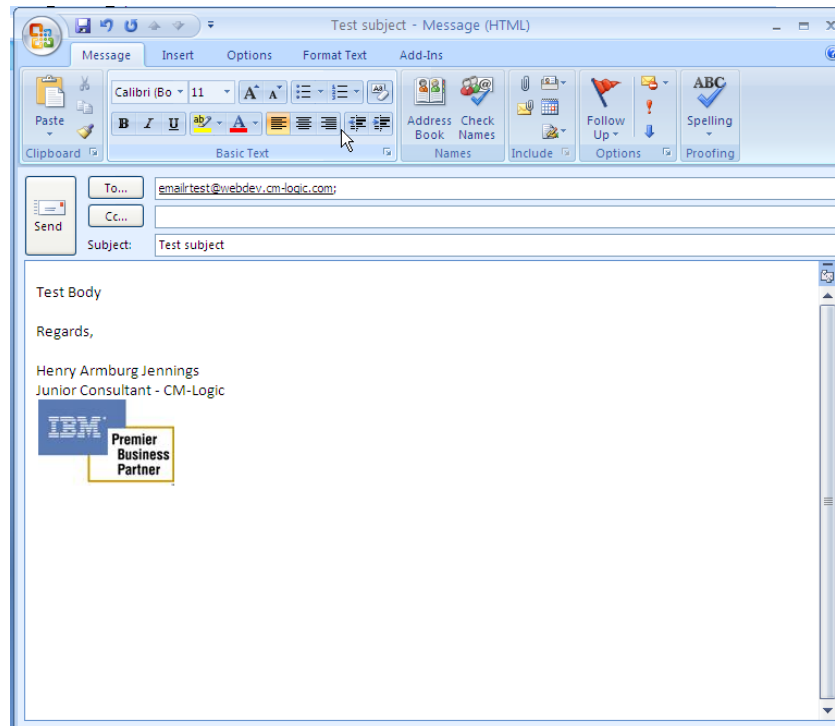


Figure 3: Test send email

Note: Check that the email arrived in the specified inbox!

7.2 Run the EmailReader.bat file

Exactly as specified in section 6.2.1. Or if it is running automatically, simply wait for it to run again.

7.3 Check Work Item in Rational Team Concert

In Rational Team Concert, in the `Team Artifacts` view, expand the relevant project and expand the `Work Items` folder. Right-click on `My Queries` and create a new query entitled 'All' with no filter. Click `Save` and run the query. Amongst the results of the query should be the 'Test Subject' Work Item. Double-click on it and see that the work item has been successfully created. It should look like Figure 4.

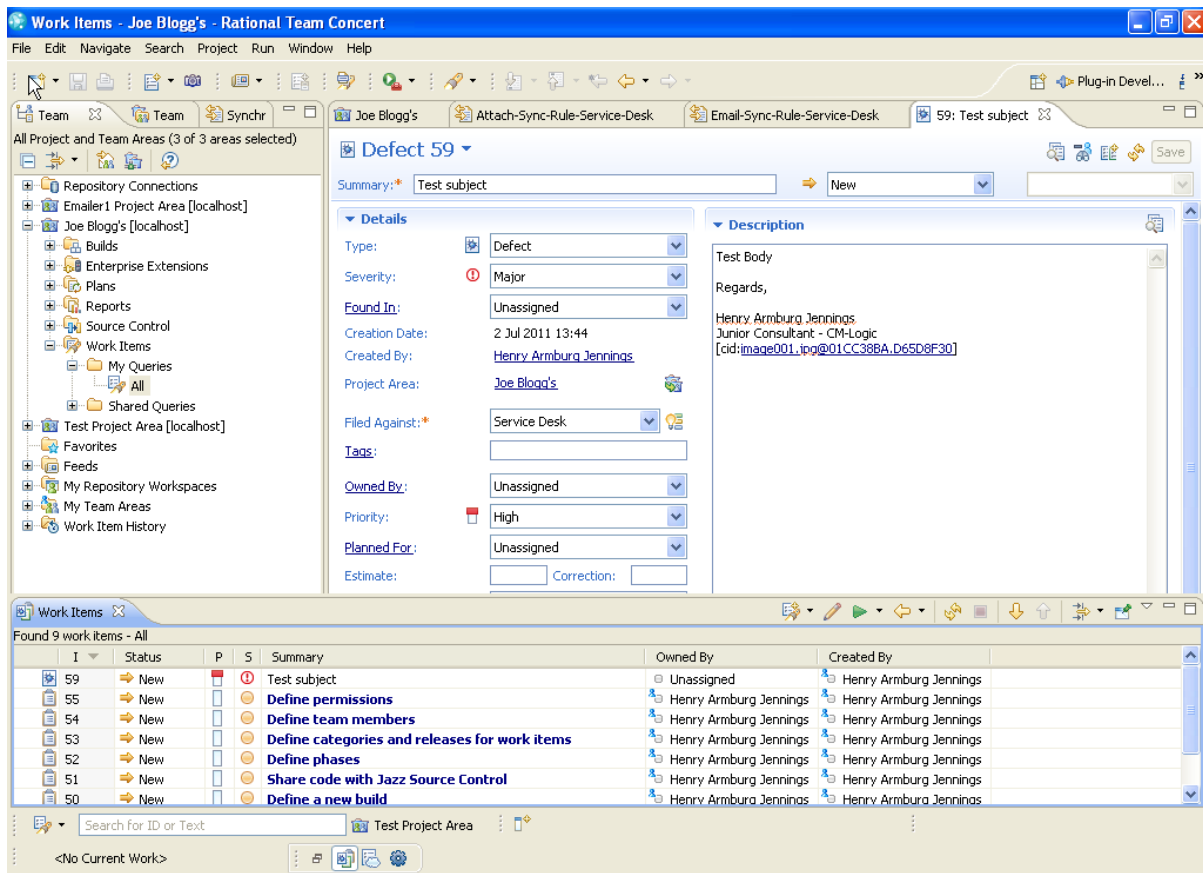


Figure 4: Work Item successfully created in RTC

8 Using the Email Reader

8.1 Creating a Work Item by Email

To create a new work item, simply send an email to the designated inbox (see section 6.1 for setting up inboxes). The subject line will be synchronised to the 'Summary' of the work item and the body of the email will be synchronised to the description of the work item.

Attributes that are configured in the emailreader.xml file in the attribute "valid-create-body-fields" can be assigned values.

For example, here, the project area, priority and severity are set for a new work item. This is done by including phrases adhering to the following syntax at the top of the body of the email (example values included):

```
Project Area=JKE Banking (Change Management)
Priority=High
Severity=Minor
```

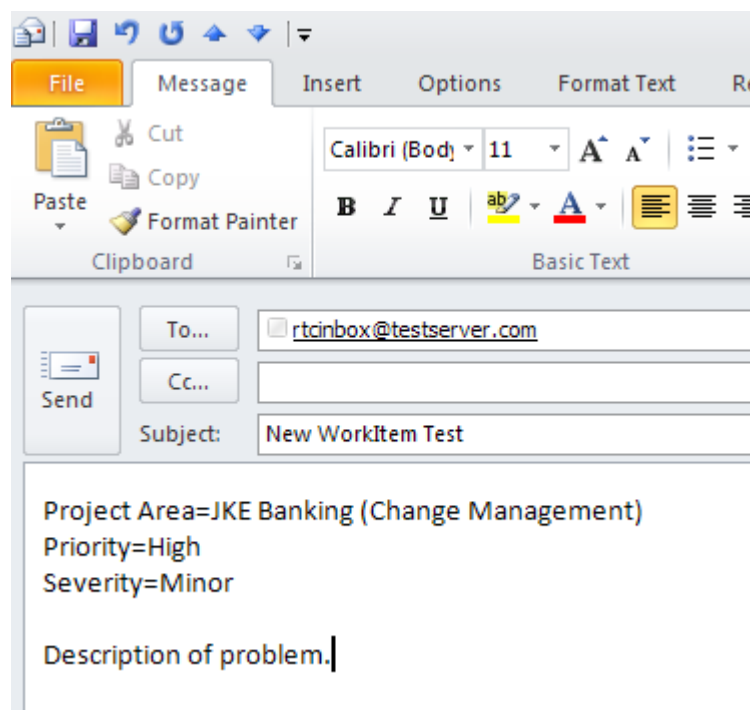


Figure 5: Setting attribute values via email

Note: If attributes are of type enumeration, for example 'Priority' or 'Severity', a valid value must be set! If an invalid value is specified, the attribute will not be populated. If the attribute is a required property (as defined in the Project Area's process configuration) then the work item will not be created.

8.2 Updating an existing Work Item by Email

To update an existing work item - for example work item 12 here - simply write [Work Item 12] anywhere in the Subject of the email. (For example, RE: [Work Item 12] would also be acceptable.)

The body of the email will be added to the work item as a new comment.

In addition to adding a new comment, attributes that are configured in the emailreader.xml file in the attribute “valid-update-body-fields” can be updated in a similar fashion as priority and severity above. For example, including:

```
Severity=Major  
Status=Resolved  
Summary=New Summary
```

would change the severity, status and summary of the work item to the above values.

Note: Once a work item has been assigned to a particular project area, it cannot be reassigned so you cannot change Project Area here. This also applies for some other attributes such as 'Creation Date' which are set by the system and cannot be altered once set for logical reasons. For a list of attributes see 10.1 Attributes set by the system on page 29.

8.3 Email Usage Example 1

SUBJECT -

Bug in GUI of Login Screen

BODY -

There is a very important problem with the GUI of the login screen of this web application. I have attached a screenshot.

```
Severity=Major  
Priority=High
```

This email will set the following attributes:

- **Summary** - Bug in GUI of Login Screen
- **Description** - There is a very important problem with the GUI of the login screen of this web application. I have attached a screenshot.

```
Severity=Major  
Priority=High
```

- **Severity** - Major
- **Priority** - High

8.4 Email Usage Example 2

SUBJECT -

Payment page bug

BODY -

Hi,

Could the London team please have a look at a bug in the loading screen of the payment page. I've attached a screenshot.

```
category=London
projectArea=Sales Website
internalTags=website,payment,London
workItemType=defect
```

Thanks

This email will set the following attributes:

- **Summary** - Payment page bug
- **Description** - Hi,

Could the London team please have a look at a bug in the loading screen of the payment page. I've attached a screenshot.

```
category=London
projectArea=Sales Website
internalTags=website,payment,London
workItemType=defect
```

Thanks

- **Category** - London
- **Project Area** - Sales Website
- **Tags** - website,payment,London
- **Work Item Type** - Defect

8.5 Email Usage Example 3

SUBJECT -

RE: [Work Item 17] Bug in GUI of Login Screen

BODY -

Hi Joe,

Could you take a look at this bug ASAP please. This defect is now potentially a blocker and it needs to be fixed by the end of tomorrow at the latest!

```
owner=joe.bloggs
Priority=Critical
dueDate=11/27/2013
```

This email will update the following attributes of work item 17:

- **Owner** - Joe Bloggs (corresponding userid - joe.bloggs)
- **Priority** - Critical
- **Due Date** - 11/27/2013
- **New Comment** - Hi Joe,

Could you take a look at this bug ASAP please. This defect is now potentially a blocker and it needs to be fixed by the end of tomorrow at the latest!

```
owner=joe.bloggs
Priority=Critical
dueDate=11/27/2013
```

8.6 Updating the Status of a work item by email

To update the status of a work item by email, ensure the email subject contains the work items id and include the status in the body of the email, for example:

```
status=In Progress
```

would update the status of a new work item from “New” to “In Progress”.

Note that the value given in the email must be a valid enumeration for the status of the given work item type, and must be a valid status to follow the current status in the workflow for that work item type.

For some workflows, resolution states have an additional Resolution enumeration value. For example, in the default project settings, the defect workflow has a resolution state of “Resolved” with possible resolution values:

- Invalid
- Works for Me

- Works as Designed
- Duplicate
- Fixed
- Unresolved
- Fixed Upstream

By default, if the status is updated to “Resolved”, the default resolution is “Fixed”. To provide a different resolution action when updating the status via email, use the attribute Internal Resolution. For example:

```
status=Resolved
internalResolution=Works for Me
```

Note that the internal resolution must be provided in the same email and the status update.

In order to update the resolution status to duplicate, RTC must know the id of the duplicate work item. Therefore, the following syntax must be used for duplicate:

```
internalResolution=Duplicate:<Work Item ID>
```

For example:

```
internalResolution=Duplicate:26
```

8.7 Email Usage Example 4

SUBJECT -

RE: [Work Item 25] Duplicate Work Item

BODY -

Hi Fred,

I believe this work item is the same as work item 26.
Therefore I am marking this work item as resolved.

```
status=Resolved
internalResolution=Duplicate:26
```

This email will update the following attributes of work item 25:

- **Status** - Resolved
- **Internal Resolution** - Duplicate
- **Links** - “Duplicated by” link to work item 26 added

New Comment - Hi Fred,

I believe this work item is the same as work item 26.
Therefore I am marking this work item as resolved.

```
status=Resolved
internalResolution=Duplicate:26
```

8.8 Setting up Review/Approval by email

To set up a review, approval or verification for a work item by email ensure the subject of the email has the work item in it (as for updating a work item) e.g. "[Work Item 12]". In the body of the email, use the following syntax to set up any number of reviews, approvals and/or verifications for the given work item:

```
internalApprovals=Approval:<List of approvers>:<Due date for approval>:<Name of approval>;Review:<List of reviewers>:<Due date for review>:<Name of review>;Verification:<List of those verifying>:<Due date for verification>:<Name of verification>
```

For example, the following:

```
internalApprovals=Approval:jbloggs@Kintrax.com:2015/05/17:Sprint One Approval;
```

would set up an approval with the following attributes:

Approver – jbloggs@Kintrax.com

Due date – 17th May 2015

Approval Name – Sprint One Approval

Approver, due date and approval name are optional attributes. If you wish to not include one, simply leave that part blank in the syntax (you must include the ":"). For example, to create a review with no due date:

```
internalApprovals=Review:jbloggs@Kintrax.com::Review with no date;
```

<i>Note that the approver/reviewer/verifier must be valid user ids in RTC.</i>
--

As with other work item attributes, attributes for reviews, approvals and verifications can be updated by email. In order to update a review/approval/verification, you will need to use the name of the review/approval/verification as the identifier. The syntax to use is exactly as for setting up a review/approval/verification.

So, for example, to update the previous example, we could send:

```
internalApprovals=Review:fbloggs@Kintrax.com:21/05/2015:Review with no date;
```

This would add the reviewer fbloggs@Kintrax.com and add the due date of the approval to 21st May 2015.

The following actions are possible by email:

- Add approver/reviewer/verifier to an approval/review/verification
- Add the due date to an approval/review/verification
- Change the due date of an approval/review/verification
- Change the name of an approval/review/verification (providing that approvers/reviewers/verifiers have already been specified).

8.9 Approving/Rejecting a Work Item by email

Approvers, reviewers and verifiers can update the status of their review by email. To do so, they must know the name of the approval/review/verification. The syntax to use is:

```
markApproval=Approval:<Approval Name>:<Updated  
status>;Review:<Review Name>:<Updated  
status>;Verification:<Verification Name>:<Updated status>
```

For example, Joe Bloggs can update his status on the Sprint One Approval by:

```
markApproval=Approval:Sprint One Approval:Approved;
```

This would update his status on the approval to “Approved”.

Note that the review status must be a valid enumeration value. By default the values are: Approved, Rejected or Pending.

8.10 Email Usage Example 5

SUBJECT -

RE: [Work Item 25]

BODY -

Creating a peer review for this work item

```
internalApprovals=Review:jbloggs@Kintrax.com:21/05/2015:Peer  
Review;
```

This email will update the following approvals attributes of work item 25:

- **Reviewers** - Joe Bloggs (corresponding userid - jbloggs@Kintrax.com)
- **Review Name** - Peer Review
- **Due Date** - 21/05/2015
- **New Comment** - Creating a peer review for this work item

```
internalApprovals=Review:jbloggs@Kintrax.com:21/05/2015:Peer  
Review;
```

8.11 Email Usage Example 6

SUBJECT -

RE: [Work Item 25]

FROM – jbloggs@Kintrax.com

BODY -

Hi,

I have reviewed this work item and can confirm I am happy to approve this.

markApproval=Review:Peer Review:Approved;

Thanks,

Joe

This email will update the following approvals attributes of work item 25:

- **Joe Bloggs review status** – Approved
- **New Comment** - Hi,

I have reviewed this work item and can confirm I am happy to approve this.

markApproval=Review:Peer Review:Approved;

Thanks,

Joe

9 Appendix A – Installation Requirements Table

The following is a table to be used to help with the installation of the JazzConnect-Email Reader Edition.

* Required entries.

PARAMETER	DESCRIPTION
JazzConnect-EmailReader Location*	Location in which to install JazzConnect-Email Reader Edition i.e. C:\KINTRAX\JazzConnect-EmailReader
Jazz Server Connection*	Connection string to login to Jazz i.e. https://localhost:9443/ccm/
Jazz User*	Userid setup for JazzConnect-Email Reader Edition to access Jazz with 'Developer' license assigned. i.e. emailsync
Jazz Password*	Corresponding password
Jazz Category/ Field Against Default	Select the default (normally the project name) Category/Filed Against value you would like new Work Item in RTC to use.

10 Appendix B – Parameters

The table below provides a comprehensive list of configuration parameters available.

<u>Parameter</u>	Options	Applicable Section(s)	Description
<u>admin-email</u>	An email address	GlobalConfig, Inbox, User	This email address will be emailed if <code>failure-admin-email</code> is enabled.
<u>body-field-alias</u>	Parameter=value	GlobalConfig, Inbox, User	Sets aliases for attribute names to be used in email bodies. For example, <code>pa=Project Area</code> enables <code>pa=JKE Banking (Change Management)</code> in an incoming email to set the project area.
<u>body-field-alias-only</u>	True or false	GlobalConfig, Inbox, User	When set to true only fields that are configured as aliases will be parsed in the body of the email.
<u>body-field-embed-newline</u>	True or false	GlobalConfig, Inbox, User	If set to true, <code>\n</code> can be used to embed new lines when setting attribute fields. If not set, the default value of <code>false</code> will be used.
<u>body-field-multiline</u>	True or false	GlobalConfig, Inbox, User	If set to true, multi-line attributes can be set in the body of the email. If not set, the default value of <code>false</code> will be used.
<u>body-field-multiline-delimiter</u>	A character/set of characters	GlobalConfig, Inbox, User	When setting a multi-line text field, the multi-line text must begin and end with this character/set of characters. E.g {Multi Line Text} where <code>body-field-multiline-delimiter="}"</code>
<u>body-field-remove</u>	True or false	GlobalConfig, Inbox, User	Removes the text used as body fields to update attributes from the

			body of the email and not added to the description section.
<u>body-fields-enabled</u>	True or false	GlobalConfig, Inbox, User	Sets whether the attributes/workflow of a work item can be manipulated via incoming emails.
<u>category</u>	<Category>/<Filed Against>	Inbox, User	Sets the Category/Filed Against attribute of the work item.
<u>creator</u>	<creator>	Inbox, User	Sets the Creator of the work item. This must be a valid RTC user.
<u>date-format</u>	Valid date format See: https://docs.oracle.com/javase/7/docs/api/java/text/SimpleDateFormat.html	Inbox	Gives the default format for the date. Any date body fields being created/updated via email must use this format
<u>email</u>	An email address	GlobalConfig, Inbox	Enter here the email address of the inbox you have specified for synchronisation.
<u>email-delete</u>	True or false	GlobalConfig, Inbox	If set to true, the emails in the specified inbox will be deleted once they have been processed.
<u>email-cc-field</u>	The attribute id of the 'CC Email' field. Usually "ccEmail"	GlobalConfig, Inbox, User	This tells the email reader which attribute to use as the CC Email field.
<u>email-from-accept</u>	A regular expression	GlobalConfig, Inbox, User	This tells the Email Reader which email addresses to accept emails from. If an email in the inbox does not come under the filter set here, it will be ignored.
<u>email-from-reject</u>	A regular expression	GlobalConfig, Inbox, User	This tells the Email Reader which email addresses to reject emails from. If an email in the inbox comes under the filter set here, it will be ignored.
<u>email-from-field</u>	The attribute id of the 'From Email' field. Usually "fromEmail"	GlobalConfig, Inbox, User	This tells the email reader which attribute to use as the From Email field.
<u>email-id-field</u>	The attribute id of	GlobalConfig,	This tells the email

	the 'Email ID' field. Usually "EmailId"	Inbox, User	reader which attribute to use as the email id field.
<u>email-ignore</u>	True or false	User	If true, emails from here will be disregarded and deleted from the inbox. Work items are not created from them.
<u>email-login</u>	The email login. For e.g. a username or email address.	Inbox	The login detail for the specified inbox.
<u>email-mailbox</u>	Use "Inbox" or specify it like this: Inbox/<email folder>	Inbox	The email folder which EmailReader will read from. Note: IMAP only (not POP3)
<u>email-passwd</u>	A password.	Inbox	The corresponding password for the login detail for the specified inbox.
<u>email-problem-delete</u>	True or false	GlobalConfig, Inbox, User	When set to true will delete email from the inbox even if they cannot be processed
<u>email-server</u>	For e.g "emailserver.Kintrax.com"	GlobalConfig, Inbox	The URL of the email server for the email inbox you are synchronising.
<u>email-subject-reject-delete</u>	True or false	GlobalConfig, Inbox, User	When set to true will ignore rejected emails. Ie if the subject line is rejected.
<u>email-to-field</u>	The attribute id of the 'To Email' field. Usually "toEmail"	GlobalConfig, Inbox, User	This tells the email reader which attribute to use as the To Email field.
<u>email-type</u>	Pop3 or Imap	GlobalConfig, Inbox	The email protocol used by your specified inbox.
<u>enabled</u>	True or false	GlobalConfig, Inbox, User	Overriding mechanism which determines whether the user/inbox/emailreader is switched on.
<u>from</u>	An email address or regular expression	User	If the 'from' email address of the email in the inbox matches this value or regular expression, this section's values will be used.
<u>frompat</u>	True or false	User	If the "from" parameter is in the form of a regular expression,

			state true.
<u>include-body-comment</u>	True or false	GlobalConfig, Inbox, User	If set to true, the body of the incoming email will be added to the work item as a new comment.
<u>internalPriority</u>	<priority>	Inbox, User	Sets the Priority attribute of the work item
<u>internalSeverity</u>	<severity>	Inbox, User	Sets the Severity attribute of the work item
<u>jazz-passwd</u>	A password. For e.g “emailsnc”	GlobalConfig, Inbox	The password associated with the super-sync user account.
<u>jazz-passwd-enc</u>	A string	GlobalConfig, Inbox	The encoded version of a password. A password can be encoded using the command line dispatcher detailed in Appendix C – Command Line Dispatcher.
<u>jazz-server</u>	For e.g “https://localhost:9443/ccm”	GlobalConfig, Inbox	The URL of the jazz repository.
<u>jazz-user</u>	A username. For e.g “emailsnc”	GlobalConfig, Inbox	The jazz userid of the super-sync user account.
<u>owner</u>	<owner>	Inbox, User	Sets the owner of the work item. This must be a valid RTC user.
<u>projectArea</u>	The name of the project area	GlobalConfig, Inbox, User	Sets which project area the work item is associated to.
<u>reply-failure-enabled</u>	True or false	GlobalConfig, Inbox, User	If set to true, a failure email will be sent if Work Item creation fails.
<u>reply-success-create-body</u>	Text	GlobalConfig, Inbox, User	Sets the body of the email to be sent if reply-success-enabled is set to true and the operation was Work Item creation.
<u>reply-success-enabled</u>	True or false	GlobalConfig, Inbox, User	If set to true, a confirmation email will be sent if Work Item creation succeeds.
<u>reply-success-update-body</u>	Text	GlobalConfig, Inbox, User	Sets the body of the email to be sent if reply-success-enabled is set

			to true and the operation was a Work Item update.
<u>reply-success-url</u>	True or false	GlobalConfig, Inbox, User	If set to true, confirmation emails, will include the Work Item URL of the created/updated Work Item.
<u>reporter</u>	<userid>:<name>:<email address>. E.g "jbloggs:Joe Bloggs:jbloggs@xxxxx.com"	Inbox, User	Sets the reporter of the work item. This must be a valid RTC user.
<u>required-create-body-fields</u>	A comma-separated list of field names.	GlobalConfig, Inbox, User	This field lists the fields that are required in an incoming email for the work item creation to succeed. (Body-fields-enabled must be set to true.)
<u>required-update-body-fields</u>	A comma-separated list of field names.	GlobalConfig, Inbox, User	This field lists the fields that are required in an incoming email for the work item update to succeed. (Body-fields-enabled must be set to true.)
<u>smtp-from-email</u>	An email address	GlobalConfig, Inbox, User	The 'From' email address for all success or failure notification emails.
<u>strip-replies</u>	True or false	GlobalConfig, Inbox, User	Instructs the EmailReader to try to remove email history from a "reply" email so that only the new part of the email is kept.
<u>subject-filter-accept</u>	A regular expression	GlobalConfig, Inbox, User	This filters emails according to their subject. If an email in the inbox does not come under the filter set here, it will be ignored.
<u>subject-filter-cleanse</u>	A regular expression	GlobalConfig, Inbox, User	This filter will remove characters from the email subject line. Any matching characters will be removed.
<u>subject-filter-reject</u>	A regular expression	GlobalConfig, Inbox, User	This filters emails according to their subject. If an email in the inbox comes under

			the filter set here, it will be ignored.
<u>subject-reply-pattern</u>	A regular expression	GlobalConfig, Inbox, User	Extra Subject reply prefixes to test against.
<u>tags</u>	A comma-separated list	GlobalConfig, Inbox, User	Sets the tags attribute for the work item. If they do not exist, they will be created.
<u>userid-fields</u>	A field or list of fields. For e.g "owner reporter"	GlobalConfig, Inbox	This field tells the EmailReader which fields it should treat as a userid.
<u>valid-create-body-fields</u>	A comma-separated list of field names	GlobalConfig, Inbox, User	This field lists the fields that can be set via an incoming email upon work item creation. (Body-fields-enabled must be set to true.) See also 10.1 and 10.2
<u>valid-update-body-fields</u>	A comma-separated list of field names	GlobalConfig, Inbox, User	This field lists the fields that can be set via an incoming email upon an existing work item. (Body-fields-enabled must be set to true.) See also 10.1 and 10.2
<u>wi-failure-admin-email</u>	True or false	GlobalConfig, Inbox, User	If set to true, the email address set in the admin-email parameter will be emailed if a Work Item create/update fails.
<u>workItemType</u>	The name of a work item type	Inbox, User	The work item type in RTC

10.1 Attributes set by the system

Please note the following Work Item attributes are set by the system, are not editable in RTC, and therefore cannot be configured in the xml file for logical reasons:

Parameter ID	Attribute Name
creationDate	Creation Date
creator	Created By
id	Id
modified	Modified Date
modifiedBy	Modified By
resolutionDate	Resolution Date
resolver	Resolved By

10.2 Body-fields

In order for Work Item attributes to be valid when creating or updating work items via email, they must be listed under “valid-create-body-fields” and “valid-update-body-fields” respectively in the xml config file. In addition, “body-fields enabled” must be set to true in the xml config file. The following work item attributes can be enabled:

Parameter ID	Attribute Name	Additional Information
category	Filed Against	Must be a valid category in RTC
com.ibm.team.apr.attribute.acceptance	Acceptance Test	
com.ibm.team.apr.attribute.complexity	Story Points	Must be a valid value in the complexity enumeration list
com.ibm.team.rtc.attribute.impact	Impact	Must be a valid value in the impact enumeration list
correctedEstimate	Corrected Estimate	Must be a numerical value followed by s,m,h,d or w
description	Description	
dueDate	Due Date	Must be in the format specified in date-format in the xml config file
duration	Estimate	Must be a numerical value followed by s,m,h,d or w
foundIn	Found In	Must be a category in RTC
internalApprovals	Approvals	Must be in the format: internalApprovals=Approval:userid1,userid2:2015/05/25:<NameOfReview>;Review:userid2:2015/04/20;Verification:userid1,userid3:2015/06/30:Verified Where date and name of review are optional (ensure all “:” are kept in place if not using date and/or name of review)
internalComments	Comments	
internalPriority	Priority	Must be a valid value in the Priority enumeration list
internalResolution	Resolution	Must be used in the same email as internalState. This gives the Resolution enumeration value (where applicable). E.g. internalState=Resolved internalResolution=Works as Designed If not used, the default value will be given. For resolving as Duplicate, internalResolution=Duplicate:<id of duplicate work item> which creates a duplicate link between the two work items.
internalSeverity	Severity	Must be a valid value in the Severity enumeration list
internalState	Status	Must be a valid state (not transition) in RTC

		workflow for the given work item type
internalSubscriptions	Subscribed By	Must be a valid RTC user
internalTags	Tags	
markApproval	Status of approval	<p>This will enable reviewers/approvers to update the state of their review/approval via email. Must be in the format: markApproval=Approval:<Name of Approval>:<State i.e. Approved, Rejected or Pending>;Review:<Name of Review>:<State>;Verification:<Name of Verification>:<State> For example: markApproval=Approval:Release 2 Approval:Rejected;Review:Release 1 Review:Approved</p>
owner	Owned By	Must be a valid RTC user
reporterCCEmail	Reporter CC Email	Must be a valid email address
reporterEmail	Reporter Email	Must be a valid email address
summary	Summary	
target	Planned For	Must be a valid release in RTC
timeSpent	Time Spent	Must be a numerical value followed by s,m,h,d or w. RTC deducts this value from the given value for Duration to calculate the Time Remaining attribute.

11 Appendix C – Command Line Dispatcher

This appendix catalogues all of the options available when using the EmailReader.bat command line program.

To run the program, navigate to <Email Reader installation directory>/bin and run:

```
EmailReader.bat <action>
```

The following are valid actions (not case sensitive):

Command	Description
encodepasswd	Encode/encrypt the given input string
createuser	Create the given user(s) in Jazz
dumpfields	Dump the details of a Jazz project fields/attributes
processemails	Process incoming emails and create/update WorkItems
testlicense	Test the validity of the loaded license file
ldaptest	Test the LDAP configuration and search for a user
hostdata	Displays the current host information
smtpptest	Test the current SMTP settings

11.1 Encode Password

The possible options when encoding a password are:

- [-p|--cmdlp] - Requests output to redirected to the console
- [-d|--debug] - Requests debug output
- input-string1 input-string2 ... input-stringN - List of strings to encode.

11.2 Create User

The possible options when creating a user are:

- [--useldap] - Use LDAP to search for the user
- [--mod-email <mod-email>] - Specify email notification options for newly created users (file)
- [--tpl-email <tpl-email>] - When creating a new user, a template user can be specified, which will be used to set the email notifications tab.
- --jazz-server <jazz-server> - Jazz server to log in to.
- --jazz-user <jazz-user> - Jazz server login.

- `--jazz-passwd <jazz-passwd>` - server password.
- `--ldapconf <ldapconf>` - Configuration file for LDAP (default: ldap.conf)
- `[--stubldap <stubldap>]` - Stub the LDAP search with a local property file
- `[--ldap-debug]` - Requests LDAP debug output
- `[-p|--cmdlp]` - Requests output to redirected to the console
- `[-d|--debug]` -Requests debug output
- `users1 users2 ... usersN` - List of users to create in Jazz.

11.3 Dump Fields

The possible options when dumping a field list are:

- `[--csv <csv>]` - Dump the fields/attributes to CSV
- `--jazz-project <jazz-project>` - Jazz project to search in.
- `--jazz-server <jazz-server>` - Jazz server to log in to.
- `--jazz-user <jazz-user>` - Jazz server login.
- `--jazz-passwd <jazz-passwd>` - Jazz server password.
- `[-p|--cmdlp]` - Requests output to redirected to the console
- `[-d|--debug]` - Requests debug output
- `[junk1 junk2 ... junkN]` - Hoover up unwanted command options.

11.4 Process Emails

The possible options when processing emails are:

- `[--uselldap]` - Use LDAP to search for the user
- `[--mod-email <mod-email>]` - Specify email notification options for newly created users (file)
- `[--tpl-email <tpl-email>]` - Specify a template user's email notification options for newly created users

- `[--poll <poll>]` - Run continuously polling every ? seconds.
- `(-c|--conf) <conf>` - Configuration file. (default: emailreader.xml)
- `[(-s|--subfilter) <subfilter>]` - Regular expression to use to filter email subjects
- `[-r|--usercreate]` - Creates the Jazz user if it doesn't already exist
- `[-m|--usesmtp]` - Allow email replies to be sent to users
- `[-o|--strip-replies]` - Attempt to remove original emails from WorkItem replies
- `[--tmp-loc] <tmp-loc>` - Temporary directory to store files. (default: C:/temp/)
- `[-n|--nowicreate]` - Do NOT create Work Items, simply process the emails
- `[-e|--email-debug]` - Requests email debug output
- `--smtpconf <smtpconf>` - Configuration file for SMTP (default: smtp.conf)
- `--ldapconf <ldapconf>` - Configuration file for LDAP (default: ldap.conf)
- `[--stubldap <stubldap>]` - Stub the LDAP search with a local property file
- `[--ldap-debug]` - Requests LDAP debug output
- `[-p|--cmdlp]` - Requests output to redirected to the console
- `[-d|--debug]` - Requests debug output
- `[junk1 junk2 ... junkN]` - Hoover up unwanted command options.

11.5 Test License

The possible options when testing a license are:

- `[-p|--cmdlp]` - Requests output to redirected to the console
- `[-d|--debug]` - Requests debug output
- `[junk1 junk2 ... junkN]` - Hoover up unwanted command options.

11.6 LDAP Test

The possible options when testing the LDAP settings are:

- `--ldapconf <ldapconf>` - Configuration file for LDAP (default: `ldap.conf`)
- `[--stubldap <stubldap>]` - Stub the LDAP search with a local property file
- `[--ldap-debug]` - Requests LDAP debug output
- `[-p|--cmdlp]` - Requests output to redirected to the console
- `[-d|--debug]` - Requests debug output
- `ldap-user1 ldap-user2 ... ldap-userN` - List of users to search LDAP.

11.7 Host Data

The possible options when getting host data are:

- `[(-h|--host) <host>]` - Hostname to return information for
- `[-p|--cmdlp]` - Requests output to redirected to the console
- `[-d|--debug]` - Requests debug output
- `[junk1 junk2 ... junkN]` - Hoover up unwanted command options.

11.8 SMTP Test

The possible options when getting host data are:

- `[--smtpconf <smtpconf>]` - Configuration file for SMTP (default: `smtp.conf`)
- `[-p|--cmdlp]` - Requests output to redirected to the console
- `[-d|--debug]` - Requests debug output
- `[emails1 emails2 ... emailsN]` - List of email addresses to send test message to.

12 Appendix D – LDAP Configuration

JazzConnect-Email Reader fully supports managing users by LDAP. To configure this, inside the `/conf` directory, the `ldap.conf` file needs to be configured. The necessary parameters and their uses are detailed below.

<u>Parameter</u>	<u>Description</u>
HOST_URL	Specifies the URL used to access the LDAP service. For example, <code>ldap://<your-server-name>:3269</code>
USER	Specifies the username used for authentication to gain access to the LDAP directory.
PWD	Specifies the password used for authentication to gain access to the LDAP directory.
BASE_DN	Specifies the structure of the LDAP tree where we will search for the user.
PROTOCOL	Specifies the protocol level of the LDAP server, it is usually 'ssl'.
AUTHENTICATION	Specifies the authentication level of the LDAP server, it is usually 'simple'.
CONTEXT_FACTORY	This is an internal field, please do not alter.
S_NAME	Specifies the LDAP field to get the users name
S_MAIL	Specifies the LDAP field to get the users email address
S_USERID	Specifies the LDAP field to get the users username
S_SEARCH	Specifies field to use for the search (i.e. The email address)

13 Appendix E - SMTP Configuration

JazzConnect-Email Reader provides the functionality to send confirmation emails to users when work items are successfully created/updated or if they fail. Access to an SMTP server is configured in this file. Inside the `/conf` directory, open and edit the `smtp.conf` file. The essential parameters and their uses are detailed below.

<u>Parameter</u>	<u>Description</u>
<code>mail.smtps.host</code>	The SMTP server to connect to.
<code>mail.smtps.auth</code>	If true, attempt to authenticate the user using the AUTH command. Defaults to false.
<code>mail.smtp.port</code>	The port of the SMTP server.
<code>transport</code>	SMTP/SMTPS.
<code>user</code>	The username.
<code>passwd</code>	The password for the username specified above.
<code>passwd-enc</code>	The Base-64 encoded password. Use either this or the 'passwd' parameter

More parameters may be needed to fully configure this file. If this is the case, a comprehensive list of parameters can be found at

<http://javamail.kenai.com/nonav/javadoc/com/sun/mail/smtp/package-summary.html>

Note: SMTP settings can be tested from the command line as detailed in section 11.8