



Copyrights and Trademarks

©Copyright 2003, 2004 by MatrixOne Inc.

All rights reserved.

PROPRIETARY RIGHTS NOTICE: This documentation is proprietary property of MatrixOne Inc. In accordance with the terms and conditions of the Software License Agreement between the Customer and MatrixOne, the Customer is allowed to print as many copies as necessary of documentation copyrighted by MatrixOne relating to the eMatrix software being used. This documentation shall be treated as confidential information and should be used only by employees or contractors with the Customer in accordance with the Agreement.

MatrixOne® is a registered trademark of MatrixOne, Inc.

eMatrix, Configurator Central, Engineering Central, Request Central, Software Central, Supplier Central, IconMail, ImageIcon, Primary Browser, Star Browser, State Browser, and Adaplet Technology are trademarks of MatrixOne Inc. Oracle® is a registered trademark of Oracle Corporation, Redwood City, California. Orbix® is a registered trademark of IONA Technologies PLC, Dublin, Ireland. CORBA®, Object Request Broker™, ORB™, and IIOP™ are trademarks or registered trademarks of Object Management Group, Inc. in the U.S. and in other countries. All other product names and services identified throughout this book are recognized as trademarks, registered trademarks, or service marks of their respective companies.

This product includes software developed by the Apache Software Foundation. (<http://www.apache.org/>)

This product includes software developed by the OpenLDAP™ Project for use in the openLDAP Toolkit. (<http://www.openldap.org/>)

This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit. (<http://www.openssl.org/>)

This product includes cryptographic software written by Eric Young. (eay@cryptsoft.com)

This product includes GifEncoder and ImageEncoder software written by Jef Poskanzer (jef@acme.com). Copyright ©1996,1998. All rights reserved.

MatrixOne Inc.
210 Littleton Road
Westford, MA 01886, USA
Telephone: 978-589 4000
Fax: 978-589-5700
Email: info@matrixone.com

Web Address: <http://www.matrixone.com>

DM-MX-22-10-50

The documentation that accompanies MatrixOne Applications describes the applications delivered by MatrixOne. This documentation includes readme files, online help, user guides, and administrator guides. If changes are made to an application or to the underlying framework, MatrixOne cannot ensure the accuracy of this documentation. These changes include but are not limited to: changing onscreen text, adding or removing fields on a page, making changes to the administrative objects in the schema, adding new JSPs or changing existing JSPs, changing trigger programs, changing the installation or login process, or changing the values in any properties file. For instructions on customizing the provided documentation, see the *Application Exchange Framework Guide*.

Table of Contents

| | |
|---|----|
| Introduction | 5 |
| Use of General Matrix Client Applications | 5 |
| Application Exchange Framework..... | 6 |
| Related Documentation | 6 |
| Checklist for Setting Up and Configuring Team Central..... | 8 |
| Setting Up MatrixOne Team Central | 10 |
| Adding Vaults, Companies, and Users | 12 |
| Overview | 12 |
| Schema Requirements for Successful Login | 12 |
| Determining Company Vaulting Strategies | 13 |
| Enabling Secure ID | 13 |
| Enabling Self-Registration..... | 14 |
| Removing the History Category from a Type's Category List | 15 |
| Configurable Pages..... | 16 |
| Managing RFQ Standard Content in Team Central Folders | 17 |
| Adding the Application Menu Tool..... | 18 |
| Enabling Text Search for Files | 19 |
| Unused Attributes for Advanced File Searches | 20 |
| Conducting Web Meetings in Multiple Languages..... | 21 |
| Accesses | 22 |
| Displaying a Separator for Currency and Quantity Fields | 23 |
| Allowing Use of Special Characters..... | 24 |
| Setting Up Self-Registration Email..... | 25 |
| Making the e-Vis Viewer Available to Users..... | 26 |
| Configuring Team Central Using Schema..... | 28 |
| Setting Up for Use with Sourcing Central | 29 |
| Configuring Team Central Using the Properties File..... | 30 |
| Properties You Should Not Change | 30 |
| Properties You Can Change | 30 |
| Configurable Automatically-Named Objects | 37 |
| Automated Processes..... | 38 |
| Automation for All Types..... | 38 |
| Automation for Documents..... | 39 |
| Automation for Folders and Content | 39 |
| Automation for Inbox Tasks | 40 |
| Automation for Meetings | 40 |
| Automation for Organizations..... | 41 |
| Automation for Persons..... | 42 |
| Automation for Routes | 42 |
| Automation for Workspaces | 43 |
| User External Authentication | 44 |

| | |
|--------------------|-----------|
| Index | 45 |
|--------------------|-----------|

Introduction

This Administrator guide describes information the System Administrator for the host company needs to know that is unique to setting up and configuring MatrixOne Team Central. Most information that Administrators need is common to all MatrixOne applications and is described in the *Application Exchange Framework Guide*, which is located in MATRICHOME/framework/Doc/pdf.

To perform administrative setup and configuration tasks, you need to understand some or all of the following:

- The concepts, terminology, processes, and procedures in Team Central, as described in the *Team Central User Guide*.
- The concepts, terminology, processes, and procedures in the Application Exchange Framework. Refer to the *Application Exchange Framework Guide*.
- The concepts, terminology, and processes related to Matrix Navigator, Matrix Business Modeler, Matrix System, and/or Matrix Query Language (MQL). See the following documents: *Matrix Basics*, *Matrix Navigator Guide*, *Matrix Business Modeler Guide*, *Matrix System Guide*, and *MQL Guide*.

Use of General Matrix Client Applications

Some of the instructions in this and other administrative guides require the use of a general Matrix client navigator, such as the:

- desktop version of Matrix Navigator (also known as the thick client)
- Web version of Matrix Navigator (also known as the thin client, PowerWeb, eMatrixApplet, and the Web Navigator)
- CWI (Collaborative Web Interface)
- Info Central

It's important to restrict the use of these general navigator applications to only a few specially-trained business administrators and to only the purposes described in the MatrixOne documentation. MatrixOne applications run JavaBean code that requires data to have specific characteristics and conditions. For example, objects may have to have certain relationships defined, have specific values entered for attributes, be in specific lifecycle states, or be in particular vaults. When a person works within the MatrixOne application user interface, these data conditions are met. But the general Matrix navigators are not necessarily aware of these conditions and therefore a person working within the general navigators can easily compromise data integrity.

There are situations in which the general navigators must or can be used, such as when:

- MatrixOne application features require data that cannot be created within the MatrixOne application user interface.
For example, some user profile information and template information must be created in a general navigator.
- Automated business rules and processes need to be configured, such as triggers and autonamers.
- Data needs to be investigated for troubleshooting, testing, or data conversion.

The general navigators should only be used in these situations, using the instructions provided in MatrixOne's documentation, and only by specially-trained business

administrators. Standard users of MatrixOne applications should never be allowed to work with their data in a general navigator and external customers should never be given access to a general navigator. (These restrictions do not apply to custom ADK applications that have been developed to access specific data for specific purposes. These applications are not general navigators and follow their own rules for preserving data integrity.)

Application Exchange Framework

Team Central is supported by a database schema, called the Application Exchange Framework. The Application Exchange Framework contains all of the administrative objects needed to run Team Central and other MatrixOne applications. The framework must be installed before you can install any MatrixOne application (with the exception of Info Central).

The Application Exchange Framework schema defines the object types, relationships, attributes, policies, and rules that govern the behavior of individual business objects that you create, modify, or manage using the Team Central.

When you install Team Central, the installation program adds programs and business objects that make the application work with the framework. For example, business objects are added that automatically name some types of business objects, such as Documents, and that control trigger programs that automate business processes. The application also includes JavaServer Pages (JSPs). JSPs contain HTML fragments, which create the application's user interface, and Java code that dynamically updates page content whenever the page is accessed. These JSPs reside in the application server's file system.

Related Documentation

In addition to this Administrator Guide, the following documents contain instructions for installing and using Team Central.

Administrator Documentation

- **Readme**
Each version of the application comes with an HTML readme file that contains installation instructions and information about new features, known issues, and closed issues.
- **Visio Business Process Chart**
This chart, TCProcessFlow10.vsd, shows the business process flow for the application. Administrators can use it to plan customizations.
- **Application Exchange Framework Guide**
This guide is available in html and pdf format. It is also for people in the host company who will configure the application. It describes the schema that underlies the application and how to configure it. Some understanding of Matrix applications, such as Matrix Navigator and Business Modeler is required. The pdf version is located in ematrix/doc/pdf/Framework.pdf and in MATRIXHOME/framework/Doc/pdf. To open the help system, open MATRIXHOME/framework/Doc/framework/launchhelp.htm.
- **Common Components Administrator Guide**
This guide, CommonComponentsAdmin.pdf and located in ematrix/doc/pdf, describes configuration options for features installed with common components, such as checkin/checkout, routes, and the default vault preference. These configuration options include triggers, configurable properties, and URL parameters.

User Documentation

- Online Help
After logging into the application, click the Help (?) tool in the upper right corner of any page to access online help.
- User Guide
The user guide is in pdf format and requires Acrobat Reader to view.
- Common Components Help
This help system describes how to use features that are common across MatrixOne applications, such as profile management, document management, file checkin and checkout, routes, and the like. Users can access this help system by clicking **Common Components Help** at the top of any application help page.
- AEF Help
This help system describes how to use features installed with the Application Exchange Framework, such as history pages and pages accessed from the toolbar. It also explains how to navigate through the user interface, such as how to use table pages and the Context Navigator. Users can access this help system by clicking **AEF Help** at the top of any application help page.

Checklist for Setting Up and Configuring Team Central

To implement the Team Central application, the software components that support it must be installed, set up, and configured to meet your business processes and environment. These supporting components include the Matrix core applications, the application server, and the Application Exchange Framework. The table below contains a list of supporting components that need to be installed and configured and where to get more information about each.

| Software Component | Setup or Configuration Option | Refer to: |
|--|---|---|
| Core technologies, app server, Matrix applications | Set up hardware, install and configure Matrix, install and configure Collaboration and application server. | Matrix Installation Guide |
| Application Exchange Framework | Define the name of the host company by changing the name of the “Company Name” business object. | “Defining Your Company” section of the <i>Application Exchange Framework Guide</i> |
| | Configure and customize the framework. | <i>Application Exchange Framework Guide</i> |
| Common Components | Configure checkin and checkout using properties and URL parameters. | <i>Common Components Administrator Guide</i> and <code>emxComponents.properties</code> |
| | Configure triggers installed with Common Components: <ul style="list-style-type: none"> • Turn off triggers that you do not want. • Change parameters passed to trigger programs, as needed. | <i>Common Components Administrator Guide</i> “Configuring Automatic Business Rules (Triggers)” section of the <i>Application Exchange Framework Guide</i> |
| Team Central | Set up features available with default installation: <ul style="list-style-type: none"> • Define currencies that should be available for conversion • Define unit of measure conversions • Determine method for defining part subcomponents or remove category | Setting Up MatrixOne Team Central |
| | Determine vaulting strategy and set up company vaults in the Matrix System application. In application, add companies, assign vaults, and add at least one administrative user per company. | Adding Vaults, Companies, and Users |
| | Display thousands separator for currency and quantity fields. | Displaying a Separator for Currency and Quantity Fields |
| | Specify special characters to allow. | Allowing Use of Special Characters |

| Software Component | Setup or Configuration Option | Refer to: |
|---------------------------|---|---|
| Team Central, cont. | Remove the History category for a type's category list. | Removing the History Category from a Type's Category List |
| | Set up folders to use for RFQ standard content (allow global read access). | Managing RFQ Standard Content in Team Central Folders |
| | Configure pages built using the framework's configurable components. | For a list of configurable pages, see Configurable Pages . For configuration options and instructions, see the Application Exchange Framework Guide . |
| | Determine whether some objects can be autonamed and the autoname prefix to use. | Configurable Automatically-Named Objects Automated Processes |
| | Configure Team Central schema: <ul style="list-style-type: none"> • Modify attributes • Configure policy access | Configuring Team Central Using Schema |
| | Set miscellaneous properties, such as defining alternate trees and settings for import/export data | Properties for miscellaneous features |
| | Configure triggers: <ul style="list-style-type: none"> • Turn off triggers that you do not want. • Change parameters passed to trigger programs, as needed. | Automated Processes "Configuring Automatic Business Rules (Triggers)" section of the Application Exchange Framework Guide |
| | Change onscreen text for Team Central, including internationalized text | "Changing Onscreen Text" and "Internationalizing the Framework and Applications" section of the Application Exchange Framework Guide and emxTeamCentralNetStringResource.properties files emxComponentsStringResource.properties emxFrameworkStringResource.properties |

Setting Up MatrixOne Team Central

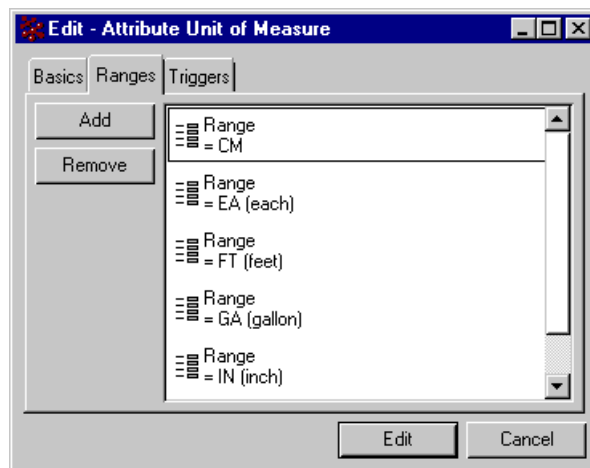
After installing Team Central, you will need to perform a few preliminary setup tasks within Matrix core applications in order for all installed features to work correctly when users log into the application. This section lists these setup tasks.

To set up Team Central

- **Define Unit of Measure Conversions.** The data for some fields shown in Team Central is based on a particular unit of measure, such as weight. The system displays this data based on the unit of measure used by the person who enters the data. On a page by page basis, users can convert data entered in English units to Metric units and vice versa. For example, if a user enters weight data using grams, another user can convert this data to pounds. Each English unit can be converted to only one Metric unit.

To define the conversion rates for English and Metric units, perform these steps for each English unit that you want users to be able to convert to a Metric unit:

- a) In Business Modeler or MQL, add and remove range values for the Unit of Measure attribute so they include all English and Metric units that you will define conversions for.



- b) Using Matrix Navigator, PowerWeb, Info Central or CWI, create a business object of Type Unit Conversion. The Name should match the English unit and the Revision should match the Metric unit. Both units must be added to the range values for the Unit of Measure attribute. Make sure the object is stored in the eService Administration vault.



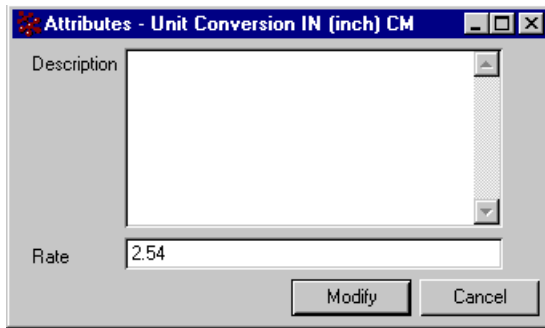
- c) In the Rate attribute for the Unit Conversion object, enter the conversion rate between the English and Metric unit.

Attributes - Unit Conversion IN (inch) CM

Description

Rate 2.54

Modify Cancel



Adding Vaults, Companies, and Users

Overview

To register a company and its employees who need to use the application, perform these steps.

1. Determine the vaulting strategy for the company using the guidelines described in *Determining Company Vaulting Strategies*.
2. If the company needs one or more vaults that are not already created, create the vault(s) using Matrix System or MQL.
3. Make sure all vaults are registered using the eService Change Administration Property wizard in Matrix Navigator. For instructions on registering objects, see the *Application Exchange Framework Guide*.
4. Using Team Central's Admin pages, create the company. Assign the company's primary vault and secondary vault as determined by the vaulting strategy. For instructions on creating companies and persons, see the Team Central User Guide or Online help.
5. Using Team Central's Admin pages, add at least one employee for each company and designate the person a Company Representative so the person can add additional employees and other profile information for the company.

If the company has secondary vaults, assign the employee's default vault according to the company's vaulting strategy. By default, a person's default vault is the same as the company's primary vault.

6. To let users assigned to different vaults create objects with the same names, set the Matrix variable `MX_CHECK_DUPLICATE_NAMES` to false.

With this variable set to false, object names must still be unique within a vault.

Schema Requirements for Successful Login

These are the schema requirements needed for a person to successfully access the Team Central application. The system fulfills all these requirements automatically when companies and employees are added using the application's Administration pages.

- To represent the person's company, a business object of type Company is created and promoted to the Active state. If the company has business units, a Business Unit object is created for each and promoted to Active. The Business Units are connected to the parent Company with the Division relationship.
- To represent the person, a person administrative object (person in Business Modeler) is defined with default settings for privileges. The person is assigned the Employee and Exchange User roles. Any employee who needs to access Team Central must be assigned the Exchange User role. If the person is designated as a Company Representative, the person is also assigned the Organization Manager role and Company Representative role. If the person will need to access Supplier Central, the person should be assigned the Buyer, Buyer Administrator, Supplier Engineer, and/or Supplier Representative roles.
- Also to represent the person, a business object of type Person is created with the same name as the administrative object. To log in, the Person business object must be in the Active state.

- The Person business object must be connected to the Company business object with the Employee relationship. If the person is assigned to a business unit, the Person business object must also be connected to the Business Unit with the Business Unit Employee relationship.
- To access the Administration pages, the Person object must be connected to the Company or Business Unit object with the Organization Representative relationship (symbolic name is relationship_CompanyRepresentative) and be assigned the Company Representative role. If the Person is connected to the host company (the Company object that was originally named Company Name), the person will have access to host company admin features, such as the ability to add companies and edit the profiles for employees of customer companies.

Determining Company Vaulting Strategies

A vault is a container for business objects created within MatrixOne applications. You assign vaults to companies and persons to control where objects created by the company's employees are stored. A company can share a vault with other companies, have its own vault, or have multiple vaults. Every person must be assigned to one of the vaults assigned to the person's company.

When assigning a vault to a company, the vault can be designated as the primary vault or as a secondary vault. If the company has only one vault, it's the primary vault. The primary vault is automatically designated as the default vault for employees. If the company has secondary vaults, the administrator adding the employee can choose a secondary vault as the person's default vault.

When a person creates a workspace, all objects created for the workspace (meetings, subscriptions, routes, etc.) are stored in the vault specified as the person's default vault, even if the objects are created by workspace members from another company (a collaboration partner). *Business objects created within a workspace are stored in the workspace creator's default vault, which is not necessarily the vault for the person who creates the object.* The only exceptions are route templates and workspace templates, which are stored in the user's company vault, not the vault for the workspace creator. This is because templates are associated with persons, not with workspaces.

For best performance, try to assign vaults to companies and persons so objects are well-distributed across vaults. Since most objects are created in the workspace creator's vault, pay particular attention to how you assign vaults to persons who are likely to create many workspaces. Don't assign all these people to the same vault.

Other MatrixOne applications installed with Team Central may have specific requirements and recommendations for company vaulting strategies. For example, when using Sourcing Central, supplier companies can share a single vault but buyer companies should not share vaults with other companies. Buyer companies can have multiple vaults. For information on an application's vaulting requirements, refer to the Administrator's Guide for that application.

Enabling Secure ID

To require specified users to enter a secure ID to log in

1. Open emxLogin.properties, which is located in ematrix/properties.
2. Set emxLogin.UseSecureId to true.

A secure ID field appears under the password field on the login page. When Company Representatives edit a person in Team Central, they can use the Login Type field to require the person to enter a secure ID when logging in.

Enabling Self-Registration

If your system is configured so the Register Here link is available from the login page, users can self-register. When this happens, the system adds the person to the database as a business object and an admin person object, and connects the person to the specified company (and business unit, if specified). But the business object for the person is in the Inactive state. The system fires a trigger that sends an email to the Company Representative for the person's company (or business unit). The email requests that the Representative activate the person. The person cannot log in until the Company Representative activates the account. For instructions on setting up the self-registration email, see [Setting Up Self-Registration Email](#).

To enable the Register Here link

1. Open `emxLogin.properties`, which is located in `ematrix/properties`.
2. Set the `emxLogin.RegisterHere` property to `true`.

Removing the History Category from a Type's Category List

The category list for many object types contains a History category. Clicking the History category presents a log of all events that have happened to the item. For each object type, the same history information is displayed for all users: buyers, suppliers, and users from inside and outside your company see the same information. If you don't want some users to see this information, you can remove the History category from the object type's category list using the following instructions. This removes the History category whenever a category list is displayed for that type of object.

For a list of all the items that have history and more information about the History page, see the "Viewing History for an Item" section of the user guide or online help.

To remove the History category from an object type's category list

1. From Business Modeler, search for the menu object that represents the type's category list. Restrict the search to only Menu objects that end with the name of the object type you want to work with.

For example, if you want to remove the History category from the Company category list, search for Menu objects with the name "*Company".

The result should contain at least one menu object named type_OBJECT, for example, type_Company.

2. Open the menu for editing.
3. Click the **Items** tab.
4. Choose the Command item that contains "History" in the name.
5. Click **Remove**.
6. Click **Edit** at the bottom of the dialog.
7. To see the change in the application, you'll need to click the **Reload Cache** tool on the toolbar, which is only available for users assigned to the Administration Manager role.

Configurable Pages

The application pages that have been built using configurable tables and forms are listed below. You edit configurable table pages using table and inquiry administrative objects and by passing parameters to `emxTable.jsp`. You edit configurable form pages using web form objects and by passing parameters to `emxForm.jsp`. For instructions on how to configure tables and forms, see the *Application Exchange Framework Guide* and the *Matrix Business Modeler Guide*.

Configurable Table Pages

These table pages have been built using AEF configurable components and therefore can be configured by editing table administrative objects and by passing parameters to `emxTable.jsp`. For instructions on how to configure tables, see the *Application Exchange Framework Guide* and the *Matrix Business Modeler Guide*.

- Routes page from My Desk
- Routes page from category list for a workspace, folder, and content
- Subroutes page for a task
- Workspace Templates page from My Desk
- All table pages within Workspace Templates category list: Access, Folders, Revision

Managing RFQ Standard Content in Team Central Folders

Standard content are documents that a buyer company requires for every RFQ that is based on a particular RFQ template. Standard content documents are typically legal documents, such as terms and conditions. Only Buyer Administrators can add and remove documents for RFQ templates.

The Buyer Administrator can add standard content to an RFQ template by adding external documents from a local disk or by adding documents that have already been added to Team Central. Typically, standard content documents are managed within a workspace(s) in Team Central.

When a buyer creates an RFQ and specifies a template, other buyers in the company and suppliers included in the RFQ should be able to view the standard content documents, even though they aren't members of the Team Central workspace that contains the document and therefore haven't been given Read access. To allow non-workspace members to view the standard content, the Team Central folders that contain standard content for RFQ templates should have the Project Security Approach setting set to global read. For instructions on enabling Global Read access for the system, see the *Team Central Administrator's Guide*. For information on assigning Global Read access to a folder, see the *Team Central User's Guide* or online help.

Here's an example of how you might work in Team Central to maintain the standard content. You could create one workspace that is specifically for standard content and create a folder that will contain standard content for each RFQ template. The people who will add the documents to the folders in Team Central will need to be given Add access to the folders. For each of these folders, set the Project Security Approach setting to Global Read.

Adding the Application Menu Tool

If your system has an application that is not available from the My Desk or Actions tabs (for example, a pre-95x application or a custom application), add the Application Menu tool to the toolbar. To access applications that are not available from the My Desk and Actions tabs, users must use the Application Menu page (emxHome.jsp). The Application Exchange Framework installs a command admin object that creates an Application Menu tool. Users can use the tool to access the Application Menu page from the main toolbar on the MatrixOne application window. To add the Application Menu tool to the main toolbar, use Business Modeler to add the AEFAppMenuToolbar command to the Toolbar menu. For detailed instructions on adding commands to menus, see the *Application Exchange Framework Guide*.

If all applications are available on the My Desk and Actions tabs (for example, your system only uses only 95x or higher, dynamic UI applications), there is no need for the Application Menu page or the Application Menu tool. When the framework installs, the tool is not included in the toolbar, so no configuration is required.

Enabling Text Search for Files

To give users the ability to search for files that have been checked into the application based on text within the files, you must install and configure Verity, Inc.'s K2 product. Contact the Verity Support Center for help on integrating with Verity K2 software. For information on configuring full text search within Matrix, see the *Matrix System Guide*.

If you will not use full text search, you may want to remove the keyword search field from the search pages by setting the following property in `emxTeamCentral.properties` to OFF.

```
emxTeamCentral.KeywordSearch=ON
```

Unused Attributes for Advanced File Searches

The advanced search for files automatically lists all non-hidden attributes for the Document type. (You reach advanced file search by clicking Search from the Team My Desk menu, clicking Find Files, and then clicking More.) The attributes listed below are not used by Team Central and therefore not used in the advanced search. If a user enters criteria for any of these attributes, the search returns no results.

- Designated User
- Version Date
- Version
- File Version
- Language

There are two solutions:

- In Business Modeler, mark the attributes as hidden so they do not appear. To do this, find the Document type and click the Indented Browser tool. For each attribute listed above that you want to hide, edit the attribute and check the Hidden box.
If you are running another MatrixOne application, such as MatrixOne Program Central, make sure it does not use these attributes. If it does, hiding them may cause problems.
- Instruct users not to enter criteria for the attributes. The online help and user guide includes instructions not to use them.

Conducting Web Meetings in Multiple Languages

Team Central can be configured to let meeting creators choose the language they want to conduct the Web meeting in. The interface for the Web meeting pages will be displayed in the chosen language for all meeting attendees. Offering Web meetings in multiple languages is currently only available when all companies use the same WebEx site. For example, you can use multiple languages if all companies use the WebEx site established by the host company.

To configure the system to support Web meetings in multiple languages

1. Make sure the Meeting Site Name and Meeting Site ID is the same for all companies that will use meetings. The Site Name should be the WebEx URL for the default language and the Site ID is the password established for the URL.
2. Open `emxTeamCentral.properties` (located in `ematrix/properties`) in a text editor.
3. Change the property `emxTeamCentral.webExLanguageEnable` to `true`.
4. Use the `emxTeamCentral.languageURL` property to define the language list and the WebEx URL to go to for each language. (This property is not used unless the `webExLanguageEnable` property is `true`.) *One of the URLs specified in the list must match the URL specified as the Meeting Site Name for each company.*

To define the list, use this syntax: `LANGUAGE,URL` for each language that should be in the list and separate the values for each language using a semicolon.

`LANGUAGE` is the key for the language as specified in the `emxTeamCentralStringResource.properties` file and `URL` is the WebEx URL for that language. For example, to list English, French, and Japanese, the `emxTeamCentral.properties` file might contain the following property. Note that the URLs given below are for example only and are not valid WebEx URLs.

```
emxTeamCentral.webExLanguageURL =
emxTeamCentral.Language.English,http://
testcc2.webex.com;emxTeamCentral.Language.Japanese,http://
/
testcc2.webex.japanese.com;emxTeamCentral.Language.French
,http://testcc2.webex.french.com
```

5. Save the file and restart the app server. If you are running a J2EE implementation, remember to run the `warutil` and deploy the archive file.

Accesses

Team Central uses these person objects to grant accesses to workspace members.

- *Workspace Member Grantor* is used to grant read access to the Workspace object only for all Workspace members regardless of specified default accesses. This lets the user create new discussions and view workspace information.
- *Workspace Lead Grantor* is used to grant accesses to the Workspace and folder objects only for all Workspace Leads.
- *Workspace Access Grantor* is used to grant accesses to all objects in the data structure in addition to the access that Workspace Member Grantor grants. These accesses are automatically granted down the data structure to at least all underlying folder and content business objects. The Discussion and Route objects for content also get the accesses from higher-level objects if their owners have not removed any accesses.
- *Route Delegation Grantor* controls temporary granted accesses in order for a user to perform delegated tasks. The delegated grantee gets the access to the Routed objects and Route that the original task assignee has.

Team Central uses simple terms to represent a group of Matrix access privileges. This table shows how general Team Central access terms map to Matrix accesses.

| Team Central Access Term | Grantor | Granted Access Privileges |
|--------------------------|--------------------------|---|
| Basic | Workspace Access Grantor | Read, but no visibility to folders. |
| Read | Workspace Access Grantor | Read, Checkout |
| Read Write | Workspace Access Grantor | Read, Checkout, Checkin, Modify, Lock, Unlock, Revise |
| Add | Workspace Access Grantor | Read, Checkout, Checkin, Modify, Lock, Unlock, Fromconnect, Toconnect, Revise |
| Remove | Workspace Access Grantor | Read, Checkout, Checkin, Modify, Lock, Unlock, Fromdisconnect, Todisconnect, Delete, Revise |
| Add Remove | Workspace Access Grantor | Read, Checkout, Checkin, Modify, Lock, Unlock, Fromconnect, Toconnect, Fromdisconnect, Todisconnect, Delete, Revise |
| Workspace Member | Workspace Member Grantor | Read |
| Workspace Lead | Workspace Lead Grantor | Read, Checkout, Checkin, Modify, Lock, Unlock, Fromconnect, Toconnect, Fromdisconnect, Todisconnect, Delete, Promote, Demote, Grant, Revoke, Revise |
| Global Read | Workspace Access Grantor | Read, Checkout, Toconnect, Todisconnect |

Displaying a Separator for Currency and Quantity Fields

You can configure the system so the currency and quantity fields display a thousandths separator (for example, 1,000).

If your system uses a period for the decimal symbol, then the thousandths separator is a comma. If your system uses a comma for a decimal, the thousandths separator is a period. The decimal symbol is set using the `emxFramework.DecimalSymbol` property in `emxSystem.properties`. This property must have the same value (a period or a comma) as the `MX_DECIMAL_SYMBOL` environment variable in `ematrix.ini`.

To show a thousandths separator in currency and quantity fields

1. Open `ematrix/properties/emxSystem.properties`.
2. Change the value for the following property to true:
`emxFramework.DigitSeparator = true`
3. Save the properties file and restart the application server. If using a J2EE implementation, run the `warutil` and `deploy` files as usual.

Allowing Use of Special Characters

You can configure the system to allow users to enter the following special characters for names for all items except companies, business units, and persons.

```
~ ( ) % & ! ; # , ' ^
```

You can configure the system to accept all special characters for descriptions, comments, and task instructions except |, #, ^, and ^>.

Users should never use special characters for companies, business units, or persons, or for any attribute whose value is used in a Tcl or MQL script.

To configure the system to allow the use of the above-listed special characters

1. In a text editor, open `emxSystem.properties`, located in `ematrix/properties`.
2. Change the `BadChars` property to it contains only the characters listed below.
Javascript Define Bad Characters for use in comment or description fields
`emxFramework.Javascript.BadChars=|_.#._.\^_.\^>_.`
3. Change the `NameBadChars` property so it contains the following characters.
Javascript Define Bad Characters for use in Name fields
`emxFramework.Javascript.NameBadChars=\\\"_._.\\'_./_.
..\\\\.\\.\\.\\.|_._.*_._.[_._.]_._.{_._}_._.=_._.<_._.>_._.@_._.$_._.?_._.\"_._.:
..|_._.§`
4. Save the property file.
5. Restart the application server. If you are running a J2EE implementation, remember to run the `warutil` and deploy the archive file.

Setting Up Self-Registration Email

The feature that emails a user's registration information to the host company requires Java mail. To enable Java mail, specific Java classes are required. The feature also requires that you specify the email address that should receive the registration information and header information for the email.

To enable Java mail and configure the registration email when using WebLogic

1. Copy the file **mail.jar** and **activation.jar** files into the following directory:
WEBLOGIC_DIRECTORY/APPLICATION_DIRECTORY/servletclasses/
2. In startWebLogic.sh/cmd, add mail.jar and activation.jar to the POST_CLASSPATH statement.
3. Add the following lines in the mimes section of the weblogic.properties file.
weblogic.httpd.mimeType.text/plain=txt, text
4. Set up the emails that will be sent when users self-register.
 - a) Open the file emxTeamCentral.properties, located in the ematrix/properties directory.
 - b) Edit these four settings so the email is sent to the appropriate email address and contains the appropriate header information.

| Property | Description |
|---|---|
| <code>String host = "mailhost.website.com";</code> | Specify the host email server. |
| <code>String from = "configurable@website.com";</code> | Specify the email address that the emails should appear to be from. |
| <code>String to = "configurable@website.com";</code> | Specify the email address the email should be sent to. This should be the email address for the person who will add the employees or who will forward the information to the person who will add the employees. |
| <code>String msgsubject = "HOST COMPANY registration request";</code> | Specify the subject of the email. |

- c) Save the file.

Making the e-Vis Viewer Available to Users

The system can be configured so people who choose e-Vis as their preferred viewer for JT files can download the installation program for e-Vis from their Profile page, which is also where they select their preferred viewer. You can also restrict the e-Vis viewer preference and download option to people who belong to a specific role. Before configuring these options, the e-Vis viewer integration must be installed and set up according to the instructions that accompany the integration. Also, the viewer must be registered and the preferred viewer selection for JT files must be configured according to the instructions in the *Application Exchange Framework Guide*.

To make the e-Vis installation program available from user Profile pages

1. Make sure you have the e-Vis installation program, called eVisClient.exe, and that you know where the file is located. You can get this installation program from <http://www.e-vis.com>.
2. In a general Matrix client application (Matrix Navigator, CWI, PowerWeb, Info Central), create a Document business object and check in the e-Vis installation file to the business object. The name, revision, and vault that you specify must match the name, revision, and vault specified in the following properties in emxTeamCentral.properties. The values shown below are the default values.

```
emxTeamCentral.eVisClientObjectName=eVisClient
emxTeamCentral.eVisClientObjectRevision=0
emxTeamCentral.eVisClientObjectVault=eService Administration
```

Currently, you must turn off triggers before creating a business object or checking in a file through Matrix core applications. To turn off triggers for the session, create a program object in Business with the command “trigger off;” and run the program in Matrix Navigator (add it to a toolset and then run it). Or issue the trigger off MQL command before creating the object in MQL or Navigator.

3. Edit the Title attribute for the Document object so it contains the name of the e-Vis installation program, which should be eVisClient.exe.
4. Configure the policy that governs the eVisClient document object so users can download the file. Use Business Modeler or MQL to configure policies.

If you will NOT restrict the Download link to a specific role:

- a) Clone the Document policy and name the new policy eVis Document policy.
- b) Remove accesses for all users except Owner, Public, and Shadow Agent.
- c) Edit the accesses for Public so it has Read, Show, and Checkout.
- d) In the general Matrix client application, assign this cloned policy to the eVisClient object that you created in Step 2 above.

Alternatively, if you will restrict the Download link to a specific role using the eVisViewerRole property, as described in the procedure that follows this one:

- a) Add the role to the access list for the Exists state of the Document policy, which is installed with the framework. Add this role to the state’s accesses even if the role is already added.
- b) Assign the Read, Checkout, and Show accesses to the role.

c) Enter the following expression for the Filter for the role's access:

```
name=='eVisClient'&&revision=='0'
```

Make sure the values you specify for the name and revision are exactly the same as the name and revision for the object you created in Step 2.

To make the e-Vis viewer available as a preferred viewer selection only to people who belong to a specific role

1. Specify the role using the `eVisViewerRole` property in `emxTeamCentral.properties`. Specify the standard name for the role, not the symbolic name.

```
emxTeamCentral.eVisViewerRole=Buyer
```

If you specify a role for this property, the e-Vis viewer will only be listed in the JT viewer preference drop-down list (on a person's profile page) for people who belong to this role. And therefore, only people of this role who choose e-Vis as their preferred viewer will have the Download link. The property accepts only one role and this role must be defined in Matrix (using Business Modeler or MQL). If no role is specified, the system lists the e-Vis viewer as a preferred JT viewer for all users (assuming the integration is installed and the system has been configured properly) and displays the Download link for all users who choose e-Vis as their preferred user.

When a role is specified for this property, the system lists this role as an available role that users can be assigned to using the administration pages. For example, suppose you define a new role in Matrix, called e-Vis Viewer Users, and specify this role for the `eVisViewerRole` property. Users with access to the administration pages can assign people to the e-Vis Viewer Users role, in addition to other roles, such as Buyer and Supplier Engineer.

2. Make sure you configure the Document policy as described in the second procedure of Step 4 above.

Configuring Team Central Using Schema

Many aspects of the application are controlled through the administrative objects installed with the Application Exchange Framework. For example, the policy that governs a business object type controls who can work with that kind of business object, what they can do, and during which states. This section describes some common configuration changes that you may want to make. All these changes can be made using Matrix Business Modeler or MQL.

When making changes to schema, remember these important points:

- When changing names of administrative objects, change only the admin object name, not the symbolic name. For more information on changing admin object names and on symbolic names, see the Application Exchange Framework Guide.*
 - Whenever you remove the Originator attribute from a type, make sure you turn off the trigger that populates the attribute by removing the Create and Revision triggers from the type.*
 - Because the application internationalizes all onscreen text, including schema names, using a string resource properties file, changing schema names does not change the names displayed onscreen. For instructions on how to change the onscreen names for internationalized schema, including policy states and attributes ranges, see the Application Exchange Framework Guide.*
-

Relationships

- **Thread**—The Thread relationship connects a Thread (called discussion in the user interface) to the object being discussed. When the object being discussed is revised, the relationship floats to the new revision. This means that all discussions are connected to the latest revision, regardless of which revision they were created for. To have the discussions stay with the revision they were created for, change the From side Revision rule from Float to None on the Thread relationship.

Setting Up for Use with Sourcing Central

If you will be using Team Central in conjunction with the Sourcing Central, make sure the following properties are set in `emxTeamCentral.properties`. For more information about each property, see [Properties You Can Change](#).

- Enable the ability to set global read access for a folder by setting `emxTeamCentral.globalReadSupport` to true.
- Enable the Packages, RFQs, and Quotations lists on the My View page by setting the property `emxTeamCentral.viewSupplierCentralApplicationItems` to true.
- If you want users to be able to add a document to more than one route, set `emxTeamCentral.DocumentsinMultipleRoutes` properties and to true.
- If you want users to be able to add documents that are locked for edit to routes, set `emxTeamCentral.CanRouteCheckoutDocuments` to true.
- If you want to prevent users from specifying image files for a company, workspace, or person, set the `emxTeamCentral.ImageCheckin` property to false.

Configuring Team Central Using the Properties File

The file called `emxTeamCentral.properties` contains properties that let you configure the content and appearance of the Team Central banner and configure many features. This file is located in `ematrix/properties`. *As with all properties files, after making changes, save the file and restart the Web server. Then make a backup copy of the file because new installations will overwrite properties files.*

You can also configure the Login and Home pages using properties files called `emxLogin.properties` and `emxSystem.properties`, also located in the `ematrix/properties` directory. This includes enabling a link that lets users self-register from the login page. For information about these files, see the *Application Exchange Framework Guide*.

Note that the `emxTeamCentral.properties` file may contain additional properties that are not listed below. These additional properties are obsolete and have no effect on the application.

Properties You Should Not Change

The `emxTeamCentral.properties` file contains several properties that should not be changed unless you are directed to do so by MatrixOne personnel. These properties are:

- The properties that control the date format for meetings.
`emxTeamCentral.CreateMeetingDateFormat= M/d/yy hh:mm:ss a zzz`
`emxTeamCentral.WebexDateFormat= M/d/yy hh:mm:ss a`
- The properties for transactions.
`emxTeamCentral.enableUpdateTransactions=true`
`emxTeamCentral.enableReadTransactions=true`
- The property that lets IconMail identify the suite.
`emxTeamCentral.ActiveSuiteName = eServiceSuiteTeamCentral`

Properties You Can Change

The properties that you can change fall into these categories:

- [Properties for miscellaneous features](#)
- [Properties for folders and routes](#)
- [Properties for My View](#)
- [Properties for meetings](#)

Properties for miscellaneous features

- **Enable 3D JT file upload**—Set the `EAIVismarkViewerEnabled` property to true to display the JtFile Checkin option on all document checkin pages. If your system has been configured for 3D JT file viewing (which includes installing the viewer and integration software, and registering the viewers and assigning formats), users can use

the link to check in JT files to folders and routes. Integrations with e-Vis Viewer and VisMockup viewers are currently supported. For instructions on registering viewers and setting them up for users, see the *Application Exchange Framework Guide*.

```
# EAI Vismark Viewer Support enabler
emxTeamCentral.EAIVismarkViewerEnabled=false
```

- **Specify the name of the object that contains the e-VisClient installation program**—When the system is configured so users can select e-Vis as a preferred viewer for JT files, a Download option is displayed on the profile page for people who select e-Vis as the preferred viewer. The link lets users download the installation program for the e-Vis viewer. These properties define the business object that has the installation program checked in. Also see [Making the e-Vis Viewer Available to Users](#).

```
emxTeamCentral.eVisClientObjectName=eVisClient
emxTeamCentral.eVisClientObjectRevision=0
emxTeamCentral.eVisClientObjectVault=eService Administration
```

- **Specify the role that should be able to choose e-Vis as a preferred viewer**—If a role is specified for this property and e-Vis has been set up as a viewer that can be selected as a preferred viewer for JT files, the system lists e-Vis as a preferred JT viewer selection only for people who are assigned to this role. If the e-Vis Download option has been configured, then the system displays the Download option for all users who choose e-Vis as their preferred viewer. If a role is not specified for the property, the system lists e-Vis as a preferred viewer selection for all users. Also see [Making the e-Vis Viewer Available to Users](#).

```
emxTeamCentral.eVisViewerRole=Buyer
```

- **Self-registration email**—Several properties control the email that contains employees' registration information when they register. These properties are only used when the suite is configured so the self-registration page does not create a user in the database. This email should be sent to the person who will add the employees or who will be responsible for forwarding the information to the appropriate Company Representative.

Host is the host email server. From is the email address that the emails should appear to be from. To is the email address the email should be sent to. This should be the email address for the person who will add the employees or who will forward the information to the person who will add the employees. Msgsubject is the subject of the email.

```
# settings for java mail when a new user registers
emxTeamCentral.host = mailhost.website.com
emxTeamCentral.from = configurable@website.com
emxTeamCentral.to = configurable@website.com
emxTeamCentral.msgsubject = Team Central registration request
```

- **Display of keyword search for files**—The KeywordSearch property lets you hide the field that lets users search for files based on text strings in the file content. If you do not have an integration with a search engine, such as Verity, you should hide the field by setting this property to OFF.

```
emxTeamCentral.KeywordSearch=ON
```

- **Define roles to be selected for persons**—Use this property to define the roles that Company Representatives can assign to persons. Only the roles listed here will be listed on the page for assigning roles to persons.

```
# ID's for the Default Roles
emxTeamCentral.DefaultRoles=role_ExchangeUser;role_SupplierEngineer;role_SupplierRepresentative;role_Buyer;role_BuyerAdministrator;role_CompanyRepresentative;
```

- **Define alternate tree menu objects for common types**—These properties define tree menu objects to be used for types that are also used in other MatrixOne applications. Types used in multiple applications, referred to as common types, include persons, companies, documents, and locations. The alternate tree menus let each application define its own tree categories for these common types. When a user views details for one of these common objects from Team Central, the system uses the tree menu object listed here instead of the standard tree menu object. (The standard tree menu object for a type has the same name as the symbolic name for the type. For example, the standard tree menu for meetings is named type_Meeting). If you want to configure the trees for these types, as described in the *Application Exchange Framework Guide*, make sure you edit the menu object named here.

The naming convention for the alternate tree properties and their values is:

```
eServiceSuiteTeamCentral.emxTreeAlternateMenuName.SYMBOLIC_NAME_OF_TYPE=TMCSYMBOLIC_NAME_OF_TYPE
```

Here are a few examples of Team Central's alternate tree properties. See the properties file for a complete list.

```
eServiceSuiteTeamCentral.emxTreeAlternateMenuName.type_Message=TMctype_Message
eServiceSuiteTeamCentral.emxTreeAlternateMenuName.type_Meeting=TMctype_Meeting
```

- **Configure vault for Part search with Engineering Central**—When Team Central is installed with Engineering Central, the property in the emxTeamCentral.properties file lets you configure the vault to use when searching for parts to add as workspace content. By default, the search is done only on the company vault of the person who is logged in.

Properties for folders and routes

- **Configure default folders and descriptions**—Properties in the emxTeamCentral.properties and emxTeamCentralStringResource.properties files let you configure the default folders and descriptions. The DefaultCategories property in emxTeamCentral.properties specifies IDs for each default folder. The IDs must be separated by semicolons and they cannot contain spaces.

```
emxTeamCentral.DefaultCategories=Production;Design;Status;Testing;Business;ChangeDocs;Prototype;RefInfo
```

In the StringResource file, the AddCategoriesDialog.ID properties specify the name for the folder and the AddCategoriesDialog.IDDesc properties specify the description. For example, here are the properties for the default folder called Production.

```
emxTeamCentral.AddCategoriesDialog.Production=Production
emxTeamCentral.AddCategoriesDialog.ProductionDesc=Current
Workspace Production Documents
```

To change the name of a default folder, change the text to the right of the equals sign for the `AddCategories.Dialog.ID` property that corresponds to the folder. To change the description for a folder, change the text to the right of the equals sign for the `AddCategories.Dialog.IDDesc` property that corresponds to the folder.

To add a new default folder, add “;ID” to the end of the `DefaultCategories` property in `emxTeamCentral.properties`, where ID is the ID for the new folder. Then add a property for the name and for the description in the `StringResource` file. For example, to add a folder called “Manuals and Readmes” with the ID “Manuals”, you would change the `emxTeamCentral.properties` setting to:

```
emxTeamCentral.DefaultCategories=Production;Design;Status;Testing;Business;ChangeDocs;Prototype;RefInfo;Manuals
```

Then you would add these properties to the `StringResource` file:

```
emxTeamCentral.AddCategoriesDialog.Manuals=Manuals and Readmes
emxTeamCentral.AddCategoriesDialog.ManualsDesc=This folder
contains files for user guides and readmes
```

- **Control whether documents can be in multiple routes**—If users should be able to add a document to more than one route, set the `DocumentsinMultipleRoutes` property to true.

```
# to allow documents in multiple Routes
emxTeamCentral.DocumentsinMultipleRoutes = true
```

- **Control whether locked documents can be routed**—To allow users to route documents that are locked for edit, set this property to true. If false, users cannot add locked documents to a route.

```
emxTeamCentral.CanRouteCheckoutDocuments = false
```

- **Enable ability to set global read access for a folder**—You can give Workspace Leads the ability to enable global read access for the folder or any of its subfolders. Global Read access should only be enabled for folders whose documents need to be viewed by people from another application, such as Sourcing Central, regardless of whether they are workspace members. For example, when using Team Central in conjunction with the Sourcing Central, Global Read access should be given to folders that will be used to store standard content documents for RFQ templates. When these documents are added to RFQ templates and then to RFQs that are based on the templates, all buyers and suppliers who have access to the RFQ will be able to view the documents. Except for these very special cases, Global Read access should not be given.

Note that from within Team Central, the standard workspace accesses apply for Global Read folders: people must have Read access to the folder to view documents.

```
# to allow Global Read Access to the documents
emxTeamCentral.globalReadSupport = false
```

- **Define the role that gets global read access**—When the `emxTeamCentral.globalReadSupport` property is true (Workspace Leads can make folders global read), use this property to define the role that is granted access to global read folders. By default the role is Global User. All users added to Team Central are assigned to the Global User role.

```
# Change the Global Read Role
emxTeamCentral.GlobalReadRole=Global User
```

- **Define the number of revisions shown for documents**—Use this property to specify how many revisions to show for documents. Use integers or All. Default is 10.

```
emxTeamCentral.DocumentHistorySize=10
```

- **Define the number of days that can be set for task due notifications** (task escalation)—Use this property to define the number of days a workspace owner can choose for task escalation notices. For example, if the property is set to 20, workspace owners can choose to send task due notifications up to 20 days before or after the task due date.

```
emxTeamCentral.SetTaskEscalationDeltaDays=20
```

- **Define the default task due time**—Use this property to specify the default time route tasks are due. The route creator can change this time as needed for each task.

```
emxTeamCentral.RouteScheduledCompletionTime = 9:00AM
```

Properties for My View

- **Control which lists to expand on My View page when users log in**—Use the following properties to specify which lists to show expanded when users first log in. To show a list expanded, set the property to true.

```
# Settings to control items to view when logged-in
emxTeamCentral.viewProjects=true
emxTeamCentral.viewTasks=false
emxTeamCentral.viewMeetings=false
emxTeamCentral.viewRoutes=false
emxTeamCentral.viewPackages=false
emxTeamCentral.viewRTS=false
emxTeamCentral.viewQuotes=false
```

- **Show user's Sourcing Central items on My View**—When the viewSupplierCentralApplicationItems property is set to true, the system displays on the My View page a Packages and RFQs list for users with the Buyer role. For users with a Supplier role (Supplier Engineer or Supplier Representative), the My View page shows a Quotation list.

```
emxTeamCentral.viewSupplierCentralApplicationItems=false
```

Properties for meetings

- **Web meetings feature available**—The enableWebEx property controls whether the meetings feature is available for the suite. If set to false, the suite will still display meetings, but the meetings will not be operable.
- **Meeting user created in WebEx**—When the validateWebExID property is true, whenever a person's meeting username or password are added or edited, the application attempts to register a person in WebEx using the specified username, password, and email address. If registration fails (for example, there may already be a

person registered with the same username and password or using the same email address), the system prompts the user for a corrected meeting username and password.

For security, WebEx requires that each account be comprised of a unique username, password, and e-mail address. For example, if you change your e-mail address in Team Central, you must set up a new WebEx account because it won't allow a different e-mail address to be used with an existing meeting username and password.

When the `validateWebExID` property is false and a person's meeting username or password is added or edited, the application updates the attributes for the Person business object in Matrix but does not attempt to validate the person's account in WebEx. The person's meeting username and password are only validated against the WebEx account when the person joins or starts a meeting.

```
emxTeamCentral.validateWebExId = true
```

- **Display of meeting information for non-Company Representatives**—The `displayMeetingsUserNameAndPasswordForUser` property controls whether non-Company Representatives can see and edit meeting username and password information.

When this property is true, non-Company Representatives can view and edit their meeting username and password. If such users attempt to access meetings and no username and password have been entered yet, a page displays that asks them to enter a username and password. When the property is set to false, the meeting username and password information is not displayed when non-Company Representatives access their profile page, even if a username and password has been entered for them. When employees who have no meeting username or password attempt to access meetings, a message displays that tells them to contact their Company Representative.

```
emxTeamCentral.displayMeetingsUserNameAndPasswordForUser = true
```

- **Display of meeting information for Company Representatives**—The `displayMeetingsUserNameAndPasswordForAdmin` property controls whether any user who is designated as a Company Representative of a customer company or customer company's business unit can view and edit meeting username and password information. When true, Company Representatives can view and edit this meeting information for any employee of their company. When false, this information is not displayed.

```
emxTeamCentral.displayMeetingsUserNameAndPasswordForAdmin = true
```

- **Display of meeting information for host company Company Representatives**—The `displayMeetingsUserNameAndPasswordForHostAdmin` property controls whether Company Representatives for the host company can view and edit meeting username and password information. When true, Company Representatives can view and edit this meeting information for any employee of any company. When false, this information is not displayed.

```
emxTeamCentral.displayMeetingsUserNameAndPasswordForHostAdmin = true
```

- **Allow meeting host to choose a language for Web meetings**—Several properties let you configure Web meetings for different languages. See [Conducting Web Meetings in Multiple Languages](#).
- **Allow users without a Web meeting username and password to join meetings**—To allow users who don't have a Web meeting username and password defined in Team Central to join meetings, set the `enableAnonymousUserInMeeting` property to

TRUE. Any meeting attendee can then join meetings as an anonymous user, without entering a username and password. The meeting creator must still have a Web meeting username and password.

```
emxTeamCentral.enableAnonymousUserInMeeting = TRUE
```

- **Define the default meeting duration**—Use this property to define the number that appears by default for meeting duration, in minutes, when a meeting host schedules a meeting. The user can change this number as needed for each meeting.

```
emxTeamCentral.MeetingDuration = 60
```

- **Assign meeting attendees from outside the workspace**—Use this property to specify whether persons other than workspace members can be added as attendees to a meeting.

```
emxTeamCentral.addMembers = true
```

- **Convert Buyer Desk members to workspace members**—Use this property to convert users who are members of the Buyer Desk to workspace members.

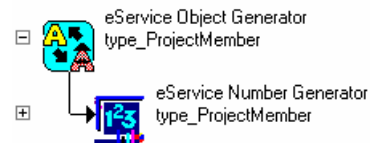
```
emxTeamCentral.ConvertBuyerDeskMembersToWorkspaceMembers =false
```

Configurable Automatically-Named Objects

Team Central automatically names these types of business objects when they are created and the autonames can be configured:

- Event
- Inbox Task
- Publish Subscribe

The application includes two business objects for each type that is named automatically: an eServices Object Generator object and an eServices Number Generator object. For example, these are the business objects that let the system name Project Member objects:



To see how the automatic names for an object type are constructed—for example, the numbering sequence, prefix, and suffix used—look at the attributes for the Object Generator that corresponds to the object type. The automatic names are suggestions only. You will probably want to configure the naming to suit your business processes. For information about how the autonaming process works and how to configure autonaming, see the “Setting Up and Configuring” chapter of the *Application Exchange Framework Guide*.

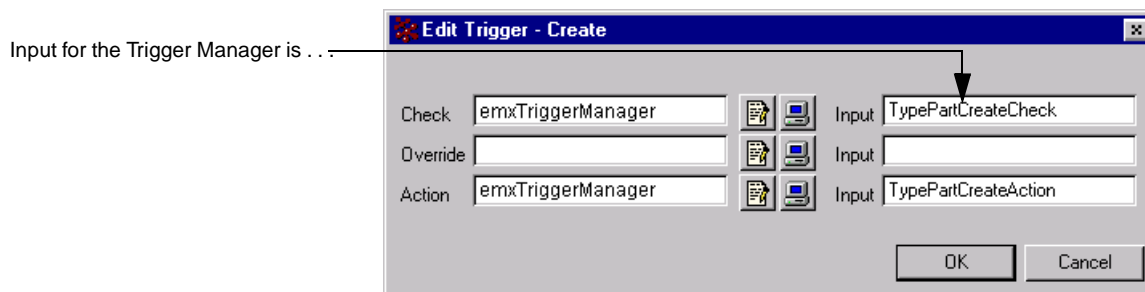
Automated Processes

This section lists the triggers that have been added to policies and other administrative objects.

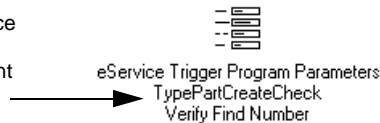
All trigger programs are run using a Trigger Manager program called `emxTriggerManager`. This Java program is specified as the Action for all triggers. The name of the eService Trigger Program Parameters business object that represents the specific trigger program to run is specified as the Input to pass to the Trigger Manager. The naming convention for the eService Trigger Program Parameters object indicates the schema object the trigger is associated with, trigger event, and the type of trigger.

[ADMIN OBJECT TYPE][ADMIN OBJECT NAME][TRIGGER EVENT][TYPE OF TRIGGER]

For example, an object named “`RelationshipVaultedDocumentsCreateAction`” indicates that the trigger is a Create Action trigger on the Vaulted Documents relationship.



. . . the name(s) of the eService Trigger Program Parameters business objects that represent trigger programs to run.



Trigger programs can be TCL programs or Java programs. Attributes on the eService Trigger Program Parameters object contain the name of the trigger program to run and parameters to pass to it. For Java programs, the object’s attributes also contain the specific method to call and constructor arguments. Commented sections within each trigger program describe the parameters accepted by the program.

For a trigger program to run, its eService Trigger Program Parameters object must be in the Active state. To turn off a trigger, demote its eService Trigger Program Parameters object to Inactive.

For more information about the Trigger Manager and eService Trigger Program Parameters objects, see the *Application Exchange Framework Guide*.

Automation for All Types

All types in the framework have two triggers that populate the Originator attribute. The diagram below shows these two triggers for the Inbox Task type. The table below the diagram describes the automation.

If you remove the Originator attribute from a type, make sure you turn off the trigger that populates the attribute by removing the Create and Revision triggers from the type.

| eService Trigger Program Parameters Object | Trigger Program and JPO Method | Description of Action |
|--|---|---|
| TypeAllCreateAction | emxcommonSetOriginator_if For information on this trigger program, see the <i>Application Exchange Framework Guide</i> . | Store user's name in Originator attribute |
| TypeAllReviseAction | emxcommonSetOriginator_if For information on this trigger program, see the <i>Application Exchange Framework Guide</i> . | Store user's name in Originator attribute |

Automation for Documents

Document objects are governed by the Document policy. Configurable automation is described in the table that follows. Also see [Automation for Folders and Content](#).

| eService Trigger Program Parameters Object | Trigger Program and JPO Method | Description of Action |
|--|---------------------------------------|---|
| TypeDocumentDeleteCheck | eServicecommonTrigUnfloatRels_if.tcl | When the latest revision of a document is deleted, this trigger fires and moves all connections from the deleted document to the previous revision. For example, if the deleted version has messages and subscriptions connected to it, these items will be connected to the previous version. This prevents losing messages and subscriptions when a version is deleted. If there is no previous version, then the connected items are lost. |
| TypeDocumentReviseAction | emxDocument floatRouteDocument method | When a new revision of a document is checked in, this trigger connects the new revision to routes that contain the previous document revision and disconnects the previous revision. The trigger makes these changes only for routes that are in the Define or In Process states. |

Automation for Folders and Content

This table lists the configurable automation related to folders and subfolders, which are called Workspace Vaults in the schema.

| eService Trigger Program Parameters Object | Trigger Program and JPO Method | Description of Action |
|--|---|---|
| RelationshipVaultedDocumentsCreateAction RelationshipVaultedDocumentsDeleteAction | eServiceUpdateCountOnConnect Disconnect.tcl | Updates the content count shown in the expandable list for folders and subfolders whenever a document or non-document (RFQs, Quotations, etc.) item is added or removed from a folder or subfolder. The count includes all subfolders under the folder. |
| TypeWorkspaceVaultGrantAction | emxWorkspaceFolderJPO grantWorkspaceFolderAccess method | When a person is granted access to a folder, this trigger grants the person access to the folder's content and its subfolders. |
| RelationshipVaultedObjectsCreateAction | emxWorkspaceConstantsJPO inheritAccess method | When an item is added to a folder, this trigger grants all accesses for the folder to the added item. |
| RelationshipVaultedObjectsDeleteAction | emxWorkspaceConstantsJPO revokeAccess method | When an item is removed from a folder, this trigger revokes granted accesses for the item. |
| RelationshipWorkspaceVaultsCreateAction | emxWorkspaceConstantsJPO inheritAccess method | When a folder is added to a workspace, this trigger grants all default workspace accesses to the folder. |
| RelationshipSubVaultsCreateAction | emxWorkspaceConstantsJPO inheritAccess method | When a subfolder is added to a folder, this trigger grants all accesses inherited from the parent folder. |
| TypeDocumentGrantAction | emxFolderContentJPO grantContentRevisionsAccess method | When a person is granted access to a document, this trigger grants access to all previous revisions of the document. |

Automation for Inbox Tasks

This table lists the automation related to Inbox Tasks, which are the tasks created for routes. Also see [Automation for Routes](#). For more information on automation for routes and tasks, see the *Application Exchange Framework Guide*.

| eService Trigger Program Parameters Object | Trigger Program and JPO Method | Description of Action |
|--|--|---|
| PolicyInboxTaskStateAssignedPromoteAction | eServicecommonTrigcCompleteTask_if.tcl | When a person completes a task, the Inbox Task object is promoted from Assigned to Complete and this trigger is fired. The trigger: <ul style="list-style-type: none"> – Enters the actual completion date. – If the approval status is Reject, stops the route and notifies the route owner. – If there are no other tasks to be completed for the route, notifies the route owner that route is complete and changes the Route status to Finished. |
| PolicyInboxTaskStateAssignedPromoteAction | | Sends a notification to the route owner if it needs review, else it promotes it to the complete state. |
| PolicyInboxTaskStateReviewPromoteAction | | Sends a notification to the task assignee informing her/him of the review completion. |
| PolicyInboxTaskStateReviewPromoteCheck | | Checks if the user is the route owner, else sends an error message to the user and prevents the promote. |
| PolicyInboxTaskStateReviewDemoteAction | | Sends a notification to the task assignee informing him of the rejection of the task. |

For email notifications to work correctly, both the sender and recipient must have valid email addresses entered in their profiles.

Automation for Meetings

This table lists the automation related to meetings.

| eService Trigger Program Parameters Object | Trigger Program and JPO Method | Description of Action |
|--|---|--|
| PolicyMeetingStateCreatePromoteAction | eServicecommonTrigaNotifyMeetingInvitation_if.tcl | When a person promotes a meeting to the Scheduled state, the system sends email to each attendee. The email lists the meeting host (person who scheduled the meeting), meeting description, and the meeting date and time. |
| PolicyMeetingStateInProgressPromoteAction | eServicecommonTrigaCleanupMeetingConnections_if.tcl | When a meeting ends, the system promotes it from In Progress to Complete. |

Automation for Organizations

This table lists the automation related to organizations business objects. The Organization policy governs Company and Business Unit objects. Whenever the term “company” is used in the automation descriptions below, the information also applies to business units.

| eService Trigger Program Parameters Object | Trigger Program and JPO Method | Description of Action |
|--|---|---|
| RelationshipCollaborationRequestCreateAction | eServicecommonTrigaNotifyCollaboration_if.tcl | When a Company Representative requests to collaborate with another company, the system connects the companies using the Collaboration Request relationship. When this relationship is created, the system fires a trigger which sends an email to: <ul style="list-style-type: none"> – all Company Representatives for the company that is being invited to collaborate; the email says there is a collaboration request from the requesting company – all Company Representatives for the company making the request; the email confirms that an email has been sent to the other company |
| RelationshipCollaborationRequestDeleteAction | eServicecommonTrigaNotifyCollaboration_if.tcl | When a Company Representative withdraws a request to collaborate with another company or the invited company rejects the invitation, the system sends an email to Company Representatives for both companies; the email says the collaboration request has been withdrawn/rejected. |
| RelationshipCollaborationPartnerCreateAction | eServicecommonTrigaNotifyCollaboration_if.tcl | When a company accepts a request to collaborate with another company, the system connects the companies with the relationship Collaboration Partner. This causes the system to fire a trigger that sends an email to: <ul style="list-style-type: none"> – all Company Representatives for the company that made the request; the email says the invited company has accepted the request – all Company Representatives for the company that accepted the request; the email confirms that the request has been accepted |
| RelationshipCollaborationPartnerDeleteAction | eServicecommonTrigaNotifyCollaboration_if.tcl | When a Company Representative dissolves a collaboration partnership, the system sends an email to: <ul style="list-style-type: none"> – all Company Representatives for the partner company; the email says the collaboration partner has been removed and specifies which company initiated the action – all Company Representatives for the company that dissolved the partnership; the email confirms that the partnership has been dissolved |
| TypeCompanyChangeNameAction | eServicecommonTrigaSyncAdminObjectName_if.tcl | When the host company changes the name of the company business object called Company Name, the trigger changes the name of the group called Company Name to match the name of the company object. |
| — | Matrix route user and route message | Change owner to Corporate. |

Automation for Persons

This table lists the automation related to persons.

| eService Trigger Program Parameters Object | Trigger Program and JPO Method | Description of Action |
|---|--|--|
| RelationshipEmployeeCreateAction | eServicecommonTrigaNotifyNewEmployee_if.tcl | When a person self-registers, an administrative person object and a person business object are created. The business object is in the Inactive state and is connected to the person's company with the Employee relationship. When the Employee relationship is created, a trigger notifies the Company Representative that the person has registered. Note that self-registration is a configurable feature that may not be enabled for your system. For more information, see the "Login Page and Properties" section of the <i>Application Exchange Framework Guide</i> . |
| PolicyPersonStateInactivePromoteAction | eServicecommonTrigaNotifyEmployeeStatus_if.tcl | When the Company Representative activates an employee, the person business object is promoted to Active. The promotion fires a trigger that notifies the employee that the account is active, so the employee can now log in. |
| AttributeEmailAddressModifyAction | eServicecommonTrigaNotifyEmailChange_if.tcl | Sends email to person whenever a change is made to their email address. The email is sent to both the old and new email address and gives the old and new address, the person who made the change, and the date and time of the change. |
| PolicyPersonStateActiveDemoteAction | eServicecommonTrigaNotifyEmployeeStatus_if.tcl | When the Company Representative deactivates an employee, the person business object is demoted to Inactive. The demotion fires a trigger that notifies the employee that the account is deactive. |
| TypePersonChangeNameAction | eServicecommonTrigaSyncAdminObjectName_if.tcl | When a person's name is changed in the application, the name of the business object for the person is changed. This name change fires a trigger that also changes the name of the administrative object for the person. |
| RelationshipCompanyRepresentativeCreateAction | eServicecommonTrigaSyncAdminGroupRole_if.tcl | When a person is assigned as a Company Representative, the system connects the person to their company or business unit with the Organization Representative relationship (symbolic name relationship_CompanyRepresentative). When this relationship is created, this trigger adds the person to the Organization Manager role. |
| RelationshipCompanyRepresentativeDeleteAction | eServicecommonTrigaSyncAdminGroupRole_if.tcl | This trigger removes a person from the Organization Manager role when they are removed as a Company Representative. |

Automation for Routes

This table lists the automation for routes. For more information on automation for routes and tasks, see the *Application Exchange Framework*. Also see [Automation for Inbox Tasks](#)

For email notifications to work correctly, both the sender and recipient must have valid email addresses entered in their profiles.

| eService Trigger Program Parameters Object | Trigger Program and JPO Method | Description of Action |
|--|--|---|
| PolicyRouteStateDefinePromoteAction | eServicecommonTrigInitiateRoute_if.tcl | When a route is started, it is promoted from Define to In Process and this trigger is fired. The trigger: <ul style="list-style-type: none"> – Creates an Inbox Task object for each person who has an Order number of 1 in the route. Copies all the information for each task (instructions, action, scheduled completion date, etc.) to the Inbox Task object. – Notifies each person via email. |
| RelationshipRouteNodeCreateAction | calls emxRouteJPO grantAccess method | When a person is added to a route, this trigger grants the person access to the route (Read Write). |
| RelationshipRouteNodeDeleteCheck | emxRouteJPO revokeAccess method | When a person is removed from a route, this trigger revokes the person's access to the route. |
| RelationshipRouteScopeCreateAction | emxRouteJPO inheritAccess method | When a route and the Route Scope relationship is created, this trigger grants Read access to the route for all Workspace Leads. |
| RelationshipRouteScopeDeleteCheck | emxWorkspaceConstantsJPO revokeAccess method | When a route and its Route Scope relationship is deleted, this trigger revokes all accesses to the route. |

Automation for Workspaces

This table shows the automation related to workspaces.

| eService Trigger Program Parameters Object | Trigger Program and JPO Method | Description of Action |
|--|---|---|
| TypeWorkspaceGrantAction | emxWorkspaceJPO grantWorkspaceAccess method | When a person is granted access for a workspace, this trigger grants the person access to the top level folder in the workspace. |
| RelationshipProjectMembersCreateAction | emxWorkspaceJPO grantMemberAccess method | When a person is added to a workspace, this trigger grants Read and Show access to the workspace. |
| TypeProjectMemberDeleteCheck | emxWorkspaceJPO revokeAllAccess method | When a person is removed from a workspace, this trigger revokes the person's granted accesses to the workspace and all children objects (folders, content, routes). |

User External Authentication

To set up external authentication for users accessing MatrixOne applications, see “Login Behavior When External Authentication is Used” in the *Installation Guide*.

If you are using Site Minder or Clear Trust for user authentication, custom JPOs and special APIs must be set up in order to authenticate users accessing lifecycles, routes, and FDA approvals. To enable custom JPO authentication, see “Enabling External Authentication” in the *Installation Guide*.

Index

Numerics

- 3D viewer
 - enabling 30

A

- ActiveSuiteName 30
- alternate tree menu objects 32
- anonymous meeting users 35
- Application Exchange Framework
 - how related to Sourcing Central 6
 - use to configure suite 5
 - user guide 6
- authentication of users 44
- automation 38
- autonomers 37

B

- banner
 - customizing 30
- business unit. See organization.

C

- CanRouteCheckoutDocuments 29, 33
- company
 - registering 12
 - vaulting 13
- company. See organization.
- CreateMeetingDateFormat 30
- currency fields 23
- customizing Team Central 30

D

- database schema 6
- date format
 - meetings 30
- DecimalSymbol property 23
- default folders 32
- default vault for part search 32
- DefaultRoles 32
- DigitSeparator 23
- disabling meetings 34
- displayMeetingsUserNameAndPasswordForAdmin 35
- displayMeetingsUserNameAndPasswordForHostAdmin 35
- displayMeetingsUserNameAndPasswordForUser 35
- Document policy 39
- documentation 6
- DocumentHistorySize 34
- documents for RTS templates 17
- DocumentsinMultipleRoutes 33
- duplicate names 12
- duration of meeting 36

E

- EAIVismarkViewerEnabled property 30
- email
 - for registration 31
- email for registration 25
- Employee role 12
- emxFramework.DecimalSymbol 23
- emxFramework.DigitSeparator 23
- emxLogin.properties 13, 14
- emxLogin.RegisterHere 14

- emxLogin.UseSecureId 13
- emxTeamCentral.GlobalReadRole 34
- emxTeamCentral.properties 30
- emxTeamCentral.RouteScheduledCompletionTime 34
- emxTreeAlternateMenuName 32
- enableAnonymousUserInMeeting 36
- enableReadTransactions 30
- enableUpdateTransactions 30
- enabling meetings 34
- eServices Number Generator 37
- eServices Object Generator 37
- event, autonamed 37
- eVis viewer 26
- eVisClientObjectName 26, 31
- eVisClientObjectRevision 26, 31
- eVisClientObjectVault 26, 31
- eVisViewerRole 27, 31
- Exchange User role 12

F

- file
 - emxTeamCentral.properties 30
- folder
 - customizing default 32
- format, date 30
- from 25
- from property 31

G

- global read access 17
- GlobalReadRole 34
- globalReadSupport 33

H

- help, online 6
- host 25
- host company
 - display of meeting information 35
- host property 31

I

- inbox task
 - automation for 40
 - autonamed 37

J

- JT files 30

K

- keyword search 31
- KeywordSearch property 19, 31

L

- languages for meetings 35
- lifecycle
 - document 39
 - inbox task 40
 - organization 41
 - organizations 41
 - person 42
 - route 42
 - task 40
 - workspace 43
- login type 13
- login, customizing 30

M

- meeting
 - adding members from outside workspace 36
- MeetingDuration 36
- meetings
 - allowing anonymous users 35
 - date format 30
 - disabling 34
 - displaying and hiding meeting username 35
 - enabling 34
 - languages 35
 - username and password 34
 - validated in WebEx 34
- message, autonamed 37
- msgsubject 25, 31
- MX_CHECK_DUPLICATE_NAMES 12

N

- naming
 - generated automatically 37

O

- online help 6
- Organization Manager role 12
- organization, automation for 41
- organizations 12
 - policy and lifecycle 41
 - status 41

P

- PDF files 6
- person
 - automation for 42
- policy
 - inbox task 40
 - organization 41
 - person 42
 - route 42
 - task 40
 - workspace 43
- Project Security Approach 17
- properties
 - for Team Central 30
 - that should not be changed 30
 - that you can change 30
- properties file
 - for configuring login and home pages 8
 - for login page 13, 14
 - for onscreen text 8
 - for suite 30
- publish subscribe, autonamed 37

Q

- quantity fields 23

R

- readme 6
- registering companies and users 12
- registration email 25, 31
- registration for user 14
- related documentation 6
- RFQ template 33
- roles 12
- route
 - automation for 42
 - autonamed 37
- RouteScheduledCompletionTime 34
- RTS template
 - standard content for 17

S

- search for keyword 31
- secure ID 13
- self-register 14
- self-registration 25, 31
- SetTaskEscalationDeltaDays 34
- setting up Sourcing Central 10
- Sourcing Central

- configuring 8
- database schema 6
- how related to AEF 6
- related documentation 6
- setting up 10
- special characters 24
- standard content 17, 33
- status
 - organizations 41

T

- task
 - automation for 40
 - autonamed 37
- to 25
- to property 31
- transactions 30
- tree menus 32
- triggers 38

U

- update transactions 30
- user authentication 44
- user guides 6
- user. See person.
- username for meetings 35
- users
 - registering 12

V

- validateWebExId 35
- validating WebEx user 34
- vaults 13
- Verity 31
- viewer
 - 3D 30
- viewMeetings 34
- viewPackages 34
- viewProjects 34
- viewQuotes 34
- viewRoutes 34
- viewRTS 34
- viewSupplierCentralApplicationItems 34
- viewTasks 34

W

- WebEx
 - enabled 34
 - information displayed 35

- user created 34
- WebexDateFormat 30
- webExLanguageEnable 21
- webExLanguageURL 21
- workspace
 - automation for 43
- workspace content
 - customizing vault for part search 32
- workspace member, autonamed 37