

Tenant Management Tool - Usage Guide

Latest TMT Version : v1.1.3

Revision History

Author	TMT Version	Document Revision Number	Date
Abdallah Hodieb	1.0.0	1.0	11 Mar. 2015
Abdallah Hodieb	1.0.0	1.1	17 Mar. 2015
Abdallah Hodieb	1.0.1	1.2	31 Mar. 2015
Abdallah Hodieb	1.1.1	1.3	7 Apr. 2015

Tenant Management Tool - Usage Guide

Common Scenarios

- [Create new empty tenant](#)
- [Create new tenant with sample data](#)
- [Make a copy of an existing tenant](#)
- [Change login-name / password of superuser](#)
- [Disable / Enable Tenant](#)
- [Change / Edit tenant properties](#)
- [Remove tenant](#)
- [Restore a removed tenant from backup](#)

Available Actions

- [List : lists all available tenants \[-l , --list \]](#)
- [List Configs : lists all config properties for a tenant \[-lc , --list-configs \]](#)
- [Create : creates a new tenant \[-c , --create \]](#)
 - [Create a new tenant with default properties](#)
 - [Create a new tenant with new path](#)
 - [Create a new tenant with new name and new path](#)
 - [Create a new tenant with new properties file](#)
 - [Create a new tenant with sample data](#)
 - [Create a new tenant with all available options](#)
- [Update Property : change tenant property \[-u , --update-property \]](#)
 - [Parameters](#)
 - [Available properties:](#)
 - [Optional Parameters](#)
 - [Update tenant property](#)
 - [Update multiple properties using a file](#)
- [Export : exports tenant \[-ex , --export \]](#)
 - [Parameters](#)
 - [Optional Parameters](#)
 - [Export tenant](#)
 - [Export tenant with data files](#)
- [Import : imports a new tenant \[-i , --import \]](#)
 - [Parameters](#)
 - [Optional Parameters](#)
 - [Import tenant with default properties](#)
 - [Import tenant with new name](#)
 - [Import tenant with new path](#)
 - [Import tenant with new name and new path](#)
 - [Import tenant with new properties file](#)
 - [Import tenant and import data files from another dir](#)
 - [Import tenant and import data files from a zip file](#)
- [Remove : removes tenant \[-r , --remove \]](#)
 - [Parameters](#)
 - [Optional Parameters](#)

Tenant Management Tool - Usage Guide

[Remove tenant](#)

[Disable : disable tenant \[-d , --disable \]](#)

[Parameters](#)

[Optional Parameters](#)

[Enable : enable tenant \[-e , --enable \]](#)

[Parameters](#)

[Optional Parameters](#)

[Change path : change tenant path \[-p , --change-path\]](#)

[Set load mode : change tenant load mode \[-slm , --set-load-mode \]](#)

[Advanced Parameters](#)

[Force : forces all actions \[-f , --force\]](#)

[Removing tenant with force mode on](#)

[Debug : prints out full stack trace for debugging \[-debug , --debug\]](#)

[Create tenant with debug mode on](#)

[JDBC Connection : specify a connection string \[-cjdbc , --config-jdbc\]](#)

[specifying a new jdbc connection](#)

[Specifying a new jdbc connection example mysql](#)

[Specifying a new jdbc connection example oracle](#)

[Server XML : specify path to server.xml \[-cxml , --config-xml\]](#)

[specifying a new jdbc connection](#)

[Disable all tenants](#)

[Enable all disabled tenants](#)

[Remove all disabled tenants \[use carefully \]](#)

[Size of all tenants \[size of path folder includes snapshots and datafiles \]](#)

[Size of all snapshot files / or data files](#)

[Change Log](#)

Common Scenarios

a list of common use patterns and how to do them quickly

Create new empty tenant

The goal is to create a new empty tenant.

- go to installation dir [usually IncortaAnalytics/Incorta_V2.0]

Linux/Mac/Windows

```
cd "Incorta/Instalation/Full/Path"
```

- If the database used is Derby [embedded database]

Linux/Mac

```
./stop.sh
```

Windows

```
stop.bat
```

- go to tmt dir

Linux/Mac/Windows

```
cd tmt
```

- Edit new_tenant.properties [skip if you want the default configs]

Linux/Mac

```
nano new_tenant.properties
```

- Create new tenant

Linux/Mac

```
./tmt.sh -c -on new_name -op new_tenant_full_path
```

Windows

```
tmt -c -on new_name -op new_tenant_full_path
```

Create new tenant with sample data

The goal is to create a new tenant and include the demo schemas and dashboards

- go to installation dir [usually IncortaAnalytics/Incorta_V2.0]

Linux/Mac/Windows

```
cd "Incorta/Instalation/Full/Path"
```

- If the database used is Derby [embedded database]

Linux/Mac

```
./stop.sh
```

Windows

```
stop.bat
```

- go to tmt dir

Linux/Mac/Windows

```
cd tmt
```

- Edit new_tenant.properties [skip if you want the default configs]

Linux/Mac

```
nano new_tenant.properties
```

- Create new tenant

Linux/Mac

```
./tmt.sh -c -s -on new_name -op new_tenant_full_path
```

Windows

```
tmt -c -s -on new_name -op new_tenant_full_path
```

Tenant Management Tool - Usage Guide

Make a copy of an existing tenant

The goal is to create a duplicate copy of an existing tenant

- go to installation dir [usually IncortaAnalytics/Incorta_V2.0]

Linux/Mac/Windows

```
cd "Incorta/Instalation/Full/Path"
```

- If the database used is Derby [embedded database]
- Linux/Mac

```
./stop.sh
```

Windows

```
stop.bat
```

- go to tmt dir

Linux/Mac/Windows

```
cd tmt
```

- Export the tenant to be duplicated

Linux/Mac

```
./tmt.sh -ex tenant_name tenant_name.zip
```

Windows

```
tmt -ex tenant_name tenant_name.zip
```

- Edit tenant_name.properties [skip if you want the same configs as the other tenant]

Linux/Mac

```
nano tenant_name.properties
```

- Import the tenant

if the data files are small enough you can use the -cf flag to import the data files

Linux/Mac

```
./tmt.sh -i tenant_name.zip -on new_name -op new_path -cf <old_tenant_pat>/data
```

Windows

```
tmt -i tenant_name.zip -on new_name -op new_path -cf <old_tenant_pat>\data
```

- Copy the data files / snapshots to the new directory manually [dont use the -cf]

Change login-name / password of superuser

6/32

Tenant Management Tool - Usage Guide

The goal is to change the login-name and password of the super user for a certain tenant.

- go to installation dir [usually IncortaAnalytics/Incorta_V2.0]

Linux/Mac/Windows

```
cd "Incorta/Instalation/Full/Path"
```

- If the database used is Derby [embedded database]

Linux/Mac

```
./stop.sh
```

Windows

```
stop.bat
```

- go to tmt dir

Linux/Mac/Windows

```
cd tmt
```

- Change superuser login-name

Linux/Mac

```
./tmt.sh -u tenant_name superuser-login new_login_name
```

Windows

```
tmt -u tenant_name superuser-login new_login_name
```

- Change superuser password

Linux/Mac

```
./tmt.sh -u tenant_name superuser-password new_password
```

Windows

```
tmt -u tenant_name superuser-password new_password
```

Disable / Enable Tenant

The goal is to disable or enable a tenant. [disabled tenants are ignored on startup and users can't login to them]

- go to installation dir [usually IncortaAnalytics/Incorta_V2.0]

Linux/Mac/Windows

```
cd "Incorta/Instalation/Full/Path"
```

- If the database used is Derby [embedded database]

Linux/Mac

```
./stop.sh
```

Windows

```
stop.bat
```

- go to tmt dir

Linux/Mac/Windows

```
cd tmt
```

- Disable new tenant

Linux/Mac

```
./tmt.sh -d tenant_name
```

Windows

```
tmt -d tenant_name
```

- Enable new tenant

Linux/Mac

```
./tmt.sh -e tenant_name
```

Windows

```
tmt -e tenant_name
```


Change / Edit tenant properties

The change multiple tenant properties at once

- go to installation dir [usually IncortaAnalytics/Incorta_V2.0]

Linux/Mac/Windows

```
cd "Incorta/Instalation/Full/Path"
```

- If the database used is Derby [embedded database]

Linux/Mac

```
./stop.sh
```

Windows

```
stop.bat
```

- go to tmt dir

Linux/Mac/Windows

```
cd tmt
```

- export the tenant properties

Linux/Mac

```
./tmt.sh -lc tenant_name > tenant.properties
```

Windows

```
tmt -d -lc tenant_name > tenant.properties
```

- edit the tenant.properties [add new properties, edit existing properties]
- Update tenant with new properties

Linux/Mac

```
./tmt.sh -u tenant_name file tenant.properties
```

Windows

```
tmt -u tenant_name file tenant.properties
```

Tenant Management Tool - Usage Guide

Remove tenant

The goal is to remove a tenant from the system.

- go to installation dir [usually IncortaAnalytics/Incorta_V2.0]

Linux/Mac/Windows

```
cd "Incorta/Instalation/Full/Path"
```

- If the database used is Derby [embedded database]

Linux/Mac

```
./stop.sh
```

Windows

```
stop.bat
```

- go to tmt dir

Linux/Mac/Windows

```
cd tmt
```

- Remove tenant

Linux/Mac

```
./tmt.sh -r tenant_name
```

Windows

```
tmt -r tenant_name
```

- Remove the tenant files manually.

Restore a removed tenant from backup

To restore a removed tenant from backup files.

- go to installation dir [usually IncortaAnalytics/Incorta_V2.0]

Linux/Mac/Windows

```
cd "Incorta/Instalation/Full/Path"
```

- If the database used is Derby [embedded database]

Linux/Mac

```
./stop.sh
```

Windows

```
stop.bat
```

- go to tmt dir

Linux/Mac/Windows

```
cd tmt
```

- Restore tenant from backup

Linux/Mac

```
./tmt.sh -i .bkp/tenant_name.bkp.zip
```

Windows

```
tmt -i .bkp\tenant_name.bkp.zip
```

Available Actions

List : lists all available tenants [-l , --list]

Optional filters [e/d enabled-disabled , s/x start-demand , p get path]

filter flags can be combined

ex : [eps : show paths of enabled tenants with load-mode = start]

ex : [dx : show disabled tenants with load-mode = demand]

List all available tenants

Linux/Mac

```
./tmt.sh --list
```

Windows

```
tmt --list
```

List all enabled tenants

[use d for disabled]

Linux/Mac

```
./tmt.sh --list e
```

Windows

```
tmt --list e
```

List all enabled tenants with load mode = start

[use x for load-mode = demand]

Linux/Mac

```
./tmt.sh --list es
```

Windows

```
tmt --list es
```

List all paths available tenants

Linux/Mac

```
./tmt.sh --list p | cat
```

List Configs : lists all config properties for a tenant [-lc , --list-configs]

Parameters [tenant_name]

List configs for a tenant

Linux/Mac

```
./tmt.sh -lc tenant_name
```

Windows

```
tmt -lc tenant_name
```

Save configs to file

Linux/Mac

```
./tmt.sh -lc tenant_name > file_name.properties
```

Windows

```
tmt -lc tenant_name > file_name.properties
```

Tenant Management Tool - Usage Guide

Create : creates a new tenant [-c , --create]

The **new_tenant.zip**, **sample.zip** , **sample_data.zip**, **new_tenant.properties** files must be available in the tmt folder.

The new tenant properties can be edited in the file **new_tenant.properties**

Parameters

name : the name of the created tenant

Optional Parameters

[-s a tenant with sample data shipped with incorta]

[-op <new_path> override path]

[-ol <new_load-mode> override load-mode]

[-opf <property_file_path> override all properties using properties file]

Note that (-on,-op,-ol) will override properties given by (-opf file) as well

Create a new tenant with default properties

Linux/Mac

```
./tmt.sh --create <name>
```

Windows

```
tmt --create <name>
```

Create a new tenant with new path

Linux/Mac

```
./tmt.sh --create <name> -op /new/path/to/tenant/folder
```

Windows

```
tmt --create <name> -op C:\new\path\to\tenant\folder
```

Create a new tenant with new name and new path

Linux/Mac

```
./tmt.sh --create -on NewName -op /new/path/to/tenant/folder
```

Windows

Tenant Management Tool - Usage Guide

```
tmt --create -on NewName -op C:\new\path\to\tenant\folder
```

Create a new tenant with new properties file
Linux/Mac

```
./tmt.sh --create -opf other_properties_file.properties
```

Windows

```
tmt --create -opf other_properties_file.properties
```

Create a new tenant with sample data
Linux/Mac

```
./tmt.sh --create -s
```

Windows

```
tmt --create -s
```

Create a new tenant with all available options
sample , override name/path/load-mode , with extra properties file
Linux/Mac

```
./tmt.sh --create -s -on new_name -op new_path -ol new_load_mode -opf properties_file
```

Windows

```
tmt --create -s -on new_name -op new_path -ol new_load_mode -opf properties_file
```

Update Property : change tenant property [-u , --update-property]

updates any tenant property

Parameters

tenant_name : name of tenant to change its path

property : the property name

15/32

Tenant Management Tool - Usage Guide

value : the new value

Available properties:

- file < properties_file > path to file containing multiple parameters to be changed
- Tenant Configs
 - name [tenant name] [rename]
 - path [tenant path] [change path]
 - load-mode [start , demand] [set load-mode]
 - description [any string]
 - enabled [true ,false] [enable/disable]
 - superuser-login [login name of super user]
 - superuser-password [password of super user]
 - cacheMaxReports [max cached reports ,default = 0 , requires server restart]
 - debug [debug mode on, default = false]
 - ehcache [ehcache xml config]
- Language Configs
 - l18nLanguage [language, default = en]
 - l18nCountry [Country, default = US]
 - l18nTextDirection [Text direction, default = ltr]
- Email Configs
 - mail.smtp.host
 - mail.smtp.socketFactory.port
 - mail.smtp.port
 - mail.smtp.socketFactory.class
 - mail.smtp.auth
 - auth.forgot.password.email
 - auth.forgot.password
 - forgot.password.email.message
 - forgot.password.email.subject
 - forgot.password.email.signature
 - sharing.notification.enabled
 - notification.email.message
- SSO configurations
 - sso-login-enable
 - sso-url

Optional Parameters

[-f skips confirmation before enable, useful for use in bash scripts for automation]

Update tenant property

Linux/Mac

```
./tmt.sh --update-property tenant_name property_name property_value
```


Tenant Management Tool - Usage Guide

Windows

```
tmt --update-property tenant_name property_name property_value
```

Update multiple properties using a file

Linux/Mac

```
./tmt.sh --update-property tenant_name file <properties_file>
```

Windows

```
tmt --update-property tenant_name file <properties_file>
```

Remove Property : remove tenant property [-rp , --remove-property]

removes tenant property.

Parameters

tenant_name : name of tenant to change its path

property : the property name

Update tenant property

Linux/Mac

```
./tmt.sh --remove-property tenant_name property_name
```

Windows

```
tmt --remove-property tenant_name property_name
```

Export : exports tenant [-ex , --export]

Exports tenant metadata into a zip file

Creates a properties file for the tenant that can be used for quick edits and import.

Optionally export the data files as well

Parameters

tenant_name: Name of the tenant to export

package_path : path to tenant zip package file [defaults to tenant_name.zip]

Optional Parameters

[-cf export data files]

Export tenant

Linux/Mac

```
./tmt.sh --export tenant_name package_name.zip
```

Windows

```
tmt --export tenant_name package_name.zip
```

Export tenant with data files

Linux/Mac

```
./tmt.sh --export tenant_name package_name.zip -cf
```

Windows

```
tmt --export tenant_name package_name.zip -cf
```

Import : imports a new tenant [-i , --import]

The imported tenant properties can be edited in the file **tenant_package.properties**

The application should have **write permissions** to the dir in which the package is available [to extract it] and the tenant path [in order to create the data and snapshots dirs]

Parameters

package_path : path to tenant zip package file

Optional Parameters

[-on <new_name> override name]

[-op <new_path> override path]

[-ol <new_load-mode> override load-mode]

[-opf <property_file_path> override all properties using properties file]

Note: (-on,-op,-ol) will override properties given by (-opf file) as well

[-cf <optional data files path> copy data files from "tenant_name"_data]

Note: (-cf) looks for data in the following way:

- provided with a path : -cf data_path -> all files in that dir will be imported.
- provided with a path to a zipfile : -cf data_path.zip -> all files in that zip will be imported.
- no path : -cf -> looks for **tenant_name.zip** or **tenant_name** and imports data files inside both of them if found.

Import tenant with default properties

Linux/Mac

```
./tmt.sh --import tenant_to_import.zip
```

Windows

```
tmt --import tenant_to_import.zip
```

Import tenant with new name

Linux/Mac

```
./tmt.sh --import tenant_to_import.zip -on NewTenantName
```

Windows

```
tmt --import tenant_to_import.zip -on NewTenantName
```

Tenant Management Tool - Usage Guide

Import tenant with new path

Linux/Mac

```
./tmt.sh --import tenant_to_import.zip -op /new/path/to/tenant/folder
```

Windows

```
tmt --import tenant_to_import.zip -op C:\new\path\to\tenant\folder
```

Import tenant with new name and new path

Linux/Mac

```
./tmt.sh --import tenant_to_import.zip -on NewName -op /new/path/to/tenant/folder
```

Windows

```
tmt --import tenant_to_import.zip -on NewName -op C:\new\path\to\tenant\folder
```

Import tenant with new properties file

Linux/Mac

```
./tmt.sh --import tenant_to_import.zip -opf <properties_file>
```

Windows

```
tmt --import tenant_to_import.zip -opf <properties_file>
```

Import tenant and its data files

data files should be in either **tenant_to_import_data/** dir or **tenant_to_import_data.zip**

Linux/Mac

```
./tmt.sh --import tenant_to_import.zip -cf
```

Windows

```
tmt --import tenant_to_import.zip -cf
```

Import tenant and import data files from another dir

Linux/Mac

Tenant Management Tool - Usage Guide

```
./tmt.sh --import tenant_to_import.zip -cf <data_files_dir>
```

Windows

```
tmt --import tenant_to_import.zip -cf <data_files_dir>
```

Import tenant and import data files from a zip file

The data files should be **flat inside the zip file** not nested inside other directories

Linux/Mac

```
./tmt.sh --import tenant_to_import.zip -cf data_files.zip
```

Windows

```
tmt --import tenant_to_import.zip -cf data_files.zip
```

Remove : removes tenant [-r , --remove]

the tenant is removed from the metadata but the datafiles and snapshots are not deleted from the file system.

The tenant is exported to a backup folder before removal, if the export process fails for any reasons (ex lack of write permissions) the tenant will not be removed.

Parameters

tenant_name : name of tenant to remove

Optional Parameters

[-f skips confirmation before removal, useful for use in bash scripts for automation]

Remove tenant

Linux/Mac

```
./tmt.sh --remove tenant_name
```

Windows

```
tmt --remove tenant_name
```

Disable : disable tenant [-d , --disable]

disables tenant without deleting it from the metadata. User can't login into a tenant while it is disabled. Tenant data is not loaded at server startup.

Parameters

tenant_name : name of tenant to disable

Optional Parameters

[-f skips confirmation before disable, useful for use in bash scripts for automation]

Linux/Mac

```
./tmt.sh --disable tenant_name
```

Windows

```
tmt --disable tenant_name
```

Enable : enable tenant [-e , --enable]

enables tenant.

Parameters

tenant_name : name of tenant to enable

Optional Parameters

[-f skips confirmation before enable, useful for use in bash scripts for automation]

Linux/Mac

```
./tmt.sh --enable tenant_name
```

Windows

```
tmt --enable tenant_name
```


Tenant Management Tool - Usage Guide

Rename : rename tenant [-re , --rename]

renames tenant.

Parameters

tenant_name : name of tenant to rename

new_name : new name of the tenant

Optional Parameters

[-f skips confirmation before enable, useful for use in bash scripts for automation]

Linux/Mac

```
./tmt.sh --rename tenant_name new_tenant_name
```

Windows

```
tmt --rename tenant_name new_tenant_name
```

Change path : change tenant path [-p , --change-path]

Change the tenant path to a new path.

The new path will not be created , it should already be there.

Parameters

tenant_name : name of tenant to change its path

new_path : new path to change to

Optional Parameters

[-f skips confirmation before enable, useful for use in bash scripts for automation]

Linux/Mac

```
./tmt.sh --change-path tenant_name /new/path/for/tenant
```

Windows

```
tmt --rename tenant_name new_tenant_name
```

Tenant Management Tool - Usage Guide

Set load mode : change tenant load mode [-slm , --set-load-mode]

change tenant load-mode

load mode can be “start” , “demand”

Parameters

tenant_name : name of tenant to change its path

load_mode : new path to change to

Optional Parameters

[-f skips confirmation before enable, useful for use in bash scripts for automation]

Linux/Mac

```
./tmt.sh --set-load-mode tenant_name load_mode
```

Windows

```
tmt --set-load-mode tenant_name load_mode
```

Advanced Parameters

Force : forces all actions [-f , --force]
ignores all confirmation prompts **use carefully**

Removing tenant with force mode on
Linux/Mac

```
./tmt.sh --remove tenant_name -f
```

Windows

```
tmt --remove tenant_name -f
```

Debug : prints out full stack trace for debugging [-debug , --debug]
prints full stack traces of all exceptions

Create tenant with debug mode on
Linux/Mac

```
./tmt.sh --debug --create
```

Windows

```
tmt --debug --create
```

JDBC Connection : specify a connection string [-cjdbc , --config-jdbc]

uses a different jdbc connection other than the one declared in the server installation

specifying a new jdbc connection

Linux/Mac

```
./tmt.sh --list -cjdbc driver_class_name jdbc_connection_string db_user_name db_pass
```

Windows

```
tmt --list -cjdbc driver_class_name jdbc_connection_string db_user_name db_pass
```

Specifying a new jdbc connection example mysql

Linux/Mac

```
./tmt.sh --list -cjdbc com.mysql.jdbc.Driver jdbc:mysql://127.0.0.1:3306/db db_user_name  
db_pass
```

Windows

```
tmt --list -cjdbc com.mysql.jdbc.Driver jdbc:mysql://127.0.0.1:3306/db db_user_name db_pass
```

Specifying a new jdbc connection example oracle

Linux/Mac

```
./tmt.sh --list -cjdbc oracle.jdbc.driver.OracleDriver jdbc:oracle:thin:@127.0.0.1:1521:db  
db_user_name db_pass
```

Windows

```
tmt --list -cjdbc oracle.jdbc.driver.OracleDriver jdbc:oracle:thin:@127.0.0.1:1521:db  
db_user_name db_pass
```

Server XML : specify path to server.xml [-cxml, --config-xml]

uses a different xml file containing JNDI configurations of the connection other than the one declared in the server installation

specifying a new jdbc connection

Linux/Mac

```
./tmt.sh --list -cxml /path/to/xml/file
```

Windows

```
tmt --list -cxml /path/to/xml/file
```


Advanced Scenarios [Linux / Mac only]

a list of advanced use patterns and how to do them quickly

Disable all tenants

```
for tenant in $(sh $PWD/'tmt.sh' -l)
do
sh $PWD/'tmt.sh' -f -d $tenant
done
```

Enable all disabled tenants

```
for tenant in $(sh $PWD/'tmt.sh' -l d)
do
sh $PWD/'tmt.sh' -f -e $tenant
done
```

Remove all disabled tenants [use carefully]

```
for tenant in $(sh $PWD/'tmt.sh' -l d)
do
sh $PWD/'tmt.sh' -f -r $tenant
done
```

Size of all tenants [size of path folder includes snapshots and datafiles]

```
for tenantPath in $(sh $PWD/'tmt.sh' -l p)
do
du -sh $tenantPath
done
```

Size of all snapshot files / or data files

```
for tenantPath in $(sh $PWD/'tmt.sh' -l p)
do
du -sh $tenantPath/snapshots
du -sh $tenantPath/data
done
```

Tenant Management Tool - Usage Guide

Change Log

Revision Number	Changes
1.2	Debug mode added
1.3	Document Reorganization. Create new tenant now requires a name. Remove property action. Update metadata action.