
TINT Deployment Guide

John Chilton

December 16, 2009

Table of Contents

Prerequisites	1
Configuration Conventions	1
Web GUI	2
Installation	2
Configuration	2
Proteomics Analytics (ProTIP)	4
Enabling	4
Configuration	5
Shared Configuration	5
Sequest Configuration	5
X! Tandem Configuration	5
OMSSA Configuration	6
Scaffold Configuration	6
RawExtract Configuration	6
ITraq Quantification Configuration	7

Prerequisites

Every configuration of TINT requires at least two pieces of software to be installed - a Java 5+ runtime and Apache Tomcat. An up-to-date Java runtime can be obtained [here](#). The latest 5.5 branch of Apache Tomcat can be obtained [here](#).

TINT requires more memory than Tomcat allows by default. Tomcat should be setup to increase the maximum amount of usable memory to at least 2 GB (or more if available). This can be done by setting the environment variable `CATALINA_OPTS` before starting Tomcat. For instance, to specify 2 GB of memory should be allowed `CATALINA_OPTS` should be set to `-Xmx2G`.

TINT is tested with Tomcat 5.5, but Tomcat 6 should also work.

Depending on which components are being deployed additional prerequisites may be necessary. These will be covered as needed in this document.

Configuration Conventions

All of the files used to configure TINT are located in subdirectories of the directory `.tropix` beneath the home directory of the user running Tomcat. If Tomcat is running as the user `mary`, then this directory will likely be `C:\Users\mary\.tropix` on Windows systems and `/home/mary/.tropix` on Linux systems. If this directory is not manually created before TINT is started, then it will be created on start up.

The configuration directories for the various components of TINT will be referred to with titles of the form `tropix.name.config.dir` which corresponds to the directory name beneath the `.tropix` directory previously described. For instance, the directory that configures the Web GUI is referred to in this

guide as `tropix.client.config.dir`. In the previous example this directory would be `C:\Users\mary\.tropix\client` on Windows systems and `/home/mary/.tropix/client` on Linux systems.

The various components of TINT discussed in this manual have different properties associated with them. These properties are generally read from a file named `deploy.properties` in the configuration directory corresponding to that component. For instance `login.timeout` is a property of the Web GUI component. To set the `login.timeout` property to 1000 one would create (or edit) a file named `deploy.properties` in the configuration directory `tropix.client.config.dir` and add the line `login.timeout=1000` to this file. These `deploy.properties` files are Java properties file. Each line should be of the form *property=value*.

Warning (Especially for Window's users): The backslash (`\`) character is an escape character in Java properties files. When a backslash should literally appear in a property value it must be escaped with another backslash. For instance, if one wanted to set the `storage.directory` property to `C:\foo\storage`, the line to add to `deploy.properties` would be `storage.directory=C:\\foo\\storage`.

Web GUI

Installation

From the TINT website download the file `tint.war`. While Tomcat is not running, place this file in the `webapps` directory of your Tomcat distribution.

Aside - The default URL for TINT will be `http://host:port/tint` (for instance `http://localhost:8080/tint`). If you wish to simplify this URL to `http://host:port/` delete the directory named `ROOT` from the `webapps` directory of the Tomcat distribution and rename `tint.war` to `ROOT.war`.

Warning About Firewalls - In addition to opening up a port in your firewall for the Tomcat container, the port 13139 should also be opened unless a TINT storage web service is going to be deployed. The default TINT deployment launches an embedded Jetty server that operates on port 13139 to handle file transfers.

Starting Tomcat now will start up TINT in its default state. In this state the TINT administrator may login to the site with the username `admin` and password `admin` and create other user accounts using the web interface. Users may upload files, create folders, and create wiki notes. Users may share these files with other users in shared folders and may perform searches.

If caGrid integration and/or access to proteomics tools is desired additional configuration is required.

Configuration

Common Configuration Properties

`modules`

If set, this property should specify a comma separated list of the TINT modules to be activated. These modules are `SHARING`, `PROTIP`, `GENETIP`, `LOCAL_SEARCH`, `REQUEST`, `CATALOG`, `GRID_SEARCH`. Likely combinations to specify are as follows.

`modules=SHARING, LOCAL_SEARCH` - This is the default setting (i.e the property used if the `modules` property is not set in `deploy.properties`). Under this configuration, users can upload and share files, but no Protip (proteomics) services are enabled.

`modules=SHARING, LOCAL_SEARCH, PROTIP` - This will enable discovery and utilization of ProTIP services.

`modules=PROTIP` - This will enable discovery and utilization of Protip services, but turn off sharing and searching between users.

Information on other modules is beyond the scope of this document. If there is interest in the functionality they provide please contact the TINT developers.

`storage.directory`

If set, this property specifies where files uploaded to and created by TINT get placed. This defaults to the files directory beneath `tropix.storage.config.dir`.

`login.timeout`

If set, this specifies the amount of time in seconds a user can remain logged in before an automatic logout is triggered. This defaults to 43200 (12 hours).

Database Properties

Without any configuration, TINT will create many H2 databases in the TINT configuration directory for establishing the metadata store, persistent tracking of jobs (both client and service), etc.... H2 is a database engine that runs inside the Java virtual machine like other products such as HSQLDB or Apache Derby.

Performance is not critical for most of these databases the Java based default databases work just fine. The one exception to this is the metadata database (due to its complexity and frequency of use). It is recommend that this be configured to target a MySQL database. The following six parameters must be set when targetting a database other than the default H2 database. These properties should be set in the `deploy.properties` file of the configuration directory `tropix.metadata.config.dir`.

`metadata.db.url`

If set, this parameter should specify the JDBC URL of the database to target. The structure of a JDBC URL varies based on the database engine being targetted. For instance, if a MySQL database named `tintmetadata` is to be targetted where MySQL is running on port 3306 of host `localhost` then the JDBC URL would be `jdbc:mysql://localhost:3306/tintmetadata`.

`metadata.db.username`

Username used to access the database.

`metadata.db.password`

Password used to access the database.

`metadata.db.dialect`

If set, this specifies the Hibernate dialect corresponding to the database engine being targetted. For MySQL this should be `org.hibernate.dialect.MySQLInnoDBDialect`. Here is a list of dialects for other database engines.

`metadata.db.hbm2ddl`

If set, this is the action performed by Hibernate at start up. Possible values are `validate`, `update`, `create`, and `create-drop`. For the typical MySQL setup this should just be set to `validate`.

`metadata.db.driver`

This is the JDBC driver class Java loads to interact with the database. For MySQL this should be set to `com.mysql.jdbc.Driver`. For databases other than MySQL and H2, a jar file containing the driver class specified here should be placed in the `shared/lib` directory of Apache Tomcat so the class is available at runtime.

For a concrete example, if the user mary wanted to configure TINT to target a MySQL database named `tintmetadata` that is stored in a MySQL container running on host `localhost` under port `3306` and she was going to connect to that host using the username `mary` and password `pass123` she would create a file under her home directory with the path `.tropix/metadata/deploy.properties`. She would then populate this file with the following contents.

```
metadata.db.url=jdbc:mysql://localhost:3306/tintmetadata
metadata.db.username=mary
metadata.db.password=pass123
metadata.db.dialect=org.hibernate.dialect.MySQLInnoDBDialect
metadata.db.hbm2ddl=validate
metadata.db.driver=com.mysql.jdbc.Driver
```

When these database settings are specified, the targetted database must be configured with the correct database structure. A MySQL script to execute to configure a database to hold TINT metadata can be found [here](#). This script can potentially be modified to apply to other database engines.

Proteomics Analytics (ProTIP)

There are currently 6 proteomics tools that may be integrated with TINT.

Sequest

<http://fields.scripps.edu/sequest/>

X! Tandem

<http://www.beavisinformatics.com/TANDEM/>

OMSSA

<http://pubchem.ncbi.nlm.nih.gov/omssa/>

Scaffold

<http://www.proteomesoftware.com/>

iTraq Quantification

This is a tool developed at the University of Minnesota that performs iTraq quantification on Sequest runs post processed through Scaffold.

RAW Extraction

This is an abstraction over two different tools for converting Thermo Finnigan RAW files into MzXML files. Either can be targetted using this component.

Enabling

Originally each of the ProTIP tools was designed as a web service, but these tools can be configured to run directly on the host of the TINT Web GUI. Each tool has a local URL that can be used to enable it. To enable one or more of these tools on the TINT Web GUI simply create the property `unindexed.service.urls` in the `deploy.properties` file of the configuration directory `tropix.client.config.dir` and add the desired URLs to this property as a comma separated list. For instance to enable Sequest and iTraq Quantification, the following line would be added to `tropix.client.config.dir/deploy.properties`:

```
unindexed.service.urls=local://Sequest, local://ITraqQuantitation
```

The URLs these these tools are `local://Sequest`, `local://XTandem`, `local://Omssa`, `local://Scaffold`, `local://ITraqQuantitation`, and `local://RawExtract`.

Each service enabled must be configured. The remainder of this section describes how to configure these services.

Configuration

The remainder of this section discuss various properties that may (or must) be set via the file `deploy.properties` in the directory `tropix.client.config.dir`.

Shared Configuration

The properties in this subsection apply to any ProTIP service that is enabled.

`queue.staging.path`

Each job must be staged before execution. This involves creating a directory for the external program to run in and setting up various input files. This property controls the directory in which these staged directories will be created. This defaults to whatever the Java system property `java.io.tmpdir` resolves to, which will likely be `CATALINA_HOME/temp` (i.e. the temp directory of your Tomcat container).

`queue.staging.clean`

This can be set to `true` or `false` and controls whether staged files are deleted after they are no longer needed. This defaults to `false`, but it can be set to `true` if there are configuration problems that need to be debugged.

Sequest Configuration

`sequest.path`

This must be set. It should be set to the fully qualified path of a Sequest executable.

`sequest.output.type`

If set, this parameter controls how Sequest logs are parsed to determine job progress. This defaults to `STANDARD_OUTPUT`, which corresponds to the output Sequest 28, if the PVM version of Sequest 27 is used this parameter should be set to `PVM_OUTPUT`.

X! Tandem Configuration

X! Tandem is distributed as part of the TINT distribution and will be configured automatically. These parameters are only necessary if you want to point TINT at a manually installed version of X! Tandem.

Ubuntu Linux Users: Due to library incompatibilities the version of X! Tandem that is bundled with ProTIP will not work out of the box on Ubuntu Linux systems. X! Tandem can be manually installed and targetted with the `xtandem.path` parameter mentioned below. The latest version of X! Tandem for various operating systems can be downloaded [here](#). Alternatively, the version of X! Tandem that is bundled with ProTIP will work in Ubuntu if a symbolic link is created via the following command `sudo ln -s /usr/lib/libexpat.so /usr/lib/libexpat.so.0` and `libstdc++` is installed. A `.deb` package for `libstdc++5` can be found [here](#). `.deb` files can be installed via the command `sudo dpkg -i /path/to/file`.

`xtandem.path`

If set, this should be set to the fully qualified path of an X! Tandem executable.

`x tandem.xsl.path`

If set, this should be set to the fully qualified path of the XSL file to apply to X! Tandem output files. Scaffold requires that the XSL file to use is the one named `tandem-style.xsl` in the `bin` directory of the X! Tandem distribution.

OMSSA Configuration

Like X! Tandem, OMSSA is distributed with TINT and will be installed automatically. Because OMSSA does not accept FASTA files directly, TINT targets a script that wraps OMSSA and the BLAST tool `formatdb`. `formatdb` is required to convert FASTA files into a format consumable by OMSSA.

There are three parameters involved in the above process that can be overridden. The wrapper script is automatically created if `omssa.path` is not set. If `omssa.path` is not set and the script is created, it will attempt to read `omssa.home` and `blast.home`. These should point to the directory which contains the `omssa` and `blast` distributions respectively. Both of these parameters are also optional and these programs will be automatically installed for your platform if needed.

`omssa.path`

If set, this should be set to the fully qualified path of the wrapper script for OMSSA and BLAST.

`omssa.home`

If set, this should be set to the fully qualified path of an OMSSA installation.

`blast.home`

If set, this should be set to the fully qualified path of a BLAST installation.

Scaffold Configuration

To integrate Scaffold with TINT, you will need to purchase a license and install ScaffoldBatch from Proteome Software. Information related to purchasing ScaffoldBatch can be found [here](#).

Be sure that ScaffoldBatch has been activated with a working key before TINT is started. This can be done by starting ScaffoldBatch from the command-line without any parameters. You will then be prompted for a license key.

`scaffold.path`

This must be set. This should be set to the fully qualified path of a Scaffold Batch executable.

RawExtract Configuration

This component can be used to wrap external programs that convert Thermo Finnigan `.RAW` files into `MzXML` files. Currently two tools are supported - `ReAdW` and `msconvert`. Both of these are part of the Transproteomics Pipeline (TPP).

`rawextract.path`

If set, this should be set to the fully qualified path of either the `msconvert.exe` or `ReAdW.exe` executable.

If this parameter is not set and TINT is running in a Windows environment, `ReAdW.exe` will be installed and targetted automatically. This requires that Thermo's Xcalibur software has been previously installed and the DLL file `XRawfile2.dll` is available to `ReAdW.exe`. For more information see [this page](#).

If this parameter is not set and TINT is running in a Linux environment, `ReAdW.exe` will be extracted into the directory `tropix.rawextract.config.dir` as well as a script that uses Wine to call

ReAdW.exe. For this script to work, Wine should be installed and Thermo's Xcalibur software should be installed inside of Wine.

The process of installing Xcalibur under Wine is likely going to vary based on the version of Wine and Xcalibur being used. At MSI, we were able to install Xcalibur 2.0.7 with the stock version of wine for Ubuntu 9.10. To install this simply install wine (`sudo apt-get install wine`), navigate to the Xcalibur 2.0.X directory of the Xcalibur CDROM (`cd /media/cdrom/Xcalibur\ 2.0.7`), launch `setup.exe` (`wine setup.exe`) and select all of the default options in the graphical installer that is launched.

ITraq Quantification Configuration

No additional configuration is necessary.