

## **IBM Cognos TM1**

TM1RunTI command line utility

### **Product Information**

This document applies to specifically to IBM Cognos TM1 Version 9.5.1 Hot Fix 14, which is a limited distribution release.

### **Copyright**

Licensed Materials - Property of IBM

© Copyright IBM Corp. 2007, 2011.

US Government Users Restricted Rights – Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

IBM, the IBM logo, ibm.com, TM1, and Cognos are trademarks or registered trademarks of International Business Machines Corp., in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the Web at [www.ibm.com/legal/copytrade.shtml](http://www.ibm.com/legal/copytrade.shtml).

Adobe, the Adobe logo, PostScript, and the PostScript logo are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, and/or other countries.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

## Table of Contents

<b>Table of Contents</b> .....	<b>2</b>
<b>1. Introduction</b> .....	<b>3</b>
1.1 Purpose .....	3
1.2 Audience.....	3
<b>2. Functional Description</b> .....	<b>4</b>
2.1 General Description.....	4
2.2 Syntax.....	4
2.3 Parameters .....	4
2.3.1 TM1RunTI Parameters.....	5
2.3.2 Connect Parameters .....	5
2.3.3 TI Parameters .....	7
2.3.4 Password Parameters.....	7
2.4 Configuration file.....	8
2.4.1 TM1RunTI section.....	8
2.4.2 Connect sections.....	9
2.4.3 Process sections .....	9
2.4.4 Example config file.....	9
2.4.5 Processing logic .....	9
2.4.6 Configuration filename and location.....	10
2.5 Return Codes and Error Messages .....	10
<b>3. Other Considerations</b> .....	<b>12</b>
3.1.1 Authentication .....	12
3.1.2 Password Security .....	12
3.1.3 Modes of Execution and Error Handling Limitations.....	12
<b>Appendix A: Definitions, Acronyms and Abbreviations</b> .....	<b>14</b>
<b>Appendix B: References</b> .....	<b>15</b>
<b>Appendix C: Help Text</b> .....	<b>16</b>

## 1. Introduction

### 1.1 Purpose

This document describes the functionality of a command line interface tool (in Windows parlance, a “console app”) that can be initiate a TM1 TurboIntegrator (TI) process. The command can be invoked from within an operating system shell, a batch script, or from within a TM1 TurboIntegrator process.

### 1.2 Audience

The intended audience for this utility is the developers of TM1 applications, particularly those building custom applications.

- This utility is of special interest in those application situations where TI process needed to be grouped in order to ensure that TI's that can run in parallel do so, and those which cannot be are serialized in the right order.
- In these situations, TM1RunTI may be used in conjunction with the tm1guardian.exe utility to achieve the desired results.

## 2. Functional Description

### 2.1 General Description

The tm1runTI.exe is a command line utility which can execute a TI process on a TM1 server.

### 2.2 Syntax

```
tm1runTI -?  
or tm1runTI -help  
or tm1runTI [<cmd_parm>...] [<ti_parm>...]
```

where <cmd\_parm> is one of:

```
-i <filespec>  
-process <string>  
-connect <string>  
<connect_parm>...
```

where <ti\_parm> is:

```
<parm_name> '=' <parm_value>
```

where <connect\_parm> is one of:

```
-adminhost <string>  
-server <string>  
-user <string>  
<password_parm>  
-AdminSvrSSLCertAuthority <filespec>  
-AdminSvrSSLCertID <id>  
-AdminSvrSSLCertRevList <filespec>  
-AdminSvrSSEExportKeyId <id>  
-ExportAdminSvrSSLCert <T|F>  
-CAMNamespace <string>
```

where <password\_parm> is one of:

```
-pwd <string>  
-passwordfile <filespec> -passwordkeyfile <filespec>
```

### 2.3 Parameters

Parameters can be either in a configuration file or passed on the command line. Command line parameters take precedence over parameters that are in the configuration file. This makes it possible to have persistent default parameters for relatively static parameters (such as adminhost and server) and to supply just the few parameters needed to either override the defaults or to provide values that are not easily defaulted, such as the user name or the TI process name.

There are four types of parameters:

1. Command parameters, which are used to specify the config file to use, which group of connection parameters to use, or which TI process to run.
2. Connection parameters, which are used to specify the servername, username and other information needed to connect to a TM1 server.
3. Password parameters, which can either be a username and plaintext password or can be a filename containing an encrypted password and associated keyfile used for decryption.
4. TI parameters, which are passed to the named TI.

Parameters specified on the command line must begin with dash (-) or slash (/). The parameter value is separated from the parameter name by a space, and the value can be specified as is or in quotes (if there are embedded spaces). For example:

```
tmlrun ti -server MyTM1Server -username John -pwd "my secret"
        ti_parm1=yes ti_parm2="my value"
```

### 2.3.1 TM1RunTI Parameters

Parameter	Value	Req'd	Description	Default
i	String	No	Path to config files	none
connect	String	No	This parameter can be used to specify a section in the config file containing parameters used to make server connections, such as user, pwd, CAMnamespace, etc.	none
process	String	No	Name of the TI process to call	none
help	n/a	No	Display help text to the command window (stdout). See Appendix C: Help text	n/a
?	n/a	No	Display a synopsis of command line parameters (as documented in 2.2 Syntax) to the command window (stdout).	n/a

### 2.3.2 Connect Parameters

Connect parameters are common between TM1 tools, and can be defined in their own section in order to enhance reused and avoid the effort and risks associated with maintaining multiple copies.

Parameter	Value	Req'd	Description	Default
adminhost	String	No	TM1 admin host	none
server	String	No	TM1 server name	none

Parameter	Value	Req'd	Description	Default
user	String	No	TM1 or Cognos Access Manager (CAM) username, depending on the type of authentication being used by the TM1 server.	none
AdminSvrSSLCertAuthority	String	No	The full path of the certificate authority file that issued the TM1 Admin Server's certificate.	none
AdminSvrSSLCertID	String	No	The name of the principal to whom the TM1 Admin Server's certificate is issued.  Note: The value of this parameter should be identical to the SSLCertificateID parameter in the Tm1admsrv.ini file.	none  API default is: tm1adminserver
AdminSvrSSLCertRevList	String	No	The full path of the certificate revocation file issued by the certificate authority that originally issued the TM1 Admin Server's certificate. A certificate revocation file will only exist in the event a certificate had been revoked.	none
ExportAdminSvrSSLCert	Boolean	No	Specifies whether you want the certificate authority certificate which originally issued the TM1 Admin Server's certificate to be exported from the Windows certificate store at runtime.  When this option is selected, you must also set a value for AdminSvrSSEExportKeyID as described below.	F
AdminSvrSSEExportKeyID	String	No	The identity key used to export the certificate authority certificate, which originally issued the TM1 Admin Server's certificate, from the certificate store.  This parameter is required only if you choose to use the	none

Parameter	Value	Req'd	Description	Default
			certificate store by setting ExportAdminSvrSSLCert=T.	
CAMNamespace	String	No	CAM namespace id (note: this not the CAM namespace name)  This value is needed only if the TM1 Server authenticates using CAM.	none

### 2.3.3 TI Parameters

These parameters are defined by the TurboIntegrator process and must be of the correct type (number or string).

TI Parameters	Value	Req'd	Description	Default
<ti_parm>	<value>	No	Provide the string or number value <value> to the parameter named <ti_parm>, which must be a valid parameter name accepted by the TI being run.	none

### 2.3.4 Password Parameters

Passwords are either provided in cleartext (not recommended) via the *pwd* parameter, or via an encrypted file provided by the *passwordfile* parameter. When a password file is used, it must have an accompanying keyfile for decryption.

TI Parameters	Value	Req'd	Description	Default
pwd	String	No	Password for the username given in the -user parameter, in clear text. For greater security, the password can be specified in an encrypted file using the passwordfile parameter.	none
passwordfile	String	No	Filename of the file containing the encrypted password for the specified user. If no path is specified, the TM1 server directory will be assumed. When this option is used, you cannot use -pwd.	none

TI Parameters	Value	Req'd	Description	Default
passwordkeyfile	String	No	If <i>passwordfile</i> is set, a key file is also required in order to decrypt the password. The password file and key file can be created using TM1Crypt tool. (See the TM1 Operations Guide.)	none

## 2.4 Configuration file

TM1RunTI can function with or without a configuration file. If a configuration file is specified, its parameters are read first. Parameters specified on the command line are then used to override those obtained from the configuration file.

When a configuration file is read, tm1runTI first obtains parameters from the section “[TM1RunTI]”. If a connect parameter is present, then parameter values will be obtained from the associated [Connect - <name>] section and used to override anything read from [TM1RunTI].

A “-connect” parameter can also be provided on the command line, and will override any connect parameters found in the config file.

The tm1runTI tool share has similar options to tm1guardian. In particular, options for server connection and authentication are the same. Consequently, the same config file can be used for both two tools, thus reducing the maintenance liability of maintaining connection information in multiple files.

The configuration file contains:

1. A single TM1RunTI section.
2. One or more sections defining the TI processes that may be run.
3. Zero or more sections defining connection parameters

All entries must start at column 1. Lines beginning with # are treated as comments.

Section names must be enclosed in square brackets [ ]. If a section name is repeated, only the first one is used.

Parameters within a section:

- cannot have blank lines between them
- can appear in any order
- are specified in keyword=value format.

Parameter values needs to be enclosed in quotes (") if they contain whitespace.

### 2.4.1 TM1RunTI section

The TM1RunTI section provides the parameter values needed to name the process to be executed and to name the server connection parameters to be used.

## 2.4.2 Connect sections

To facilitate easy maintenance for different server environments (such as development, test and production), connection parameters for each environment can be specified in separate sections. Each section is named using the prefix "Connect -" followed by a user defined name. For example:

```
[Connect - Production]
```

```
[Connect - Test]
```

```
[Connect - Development]
```

## 2.4.3 Process sections

Multiple TI process sections are permitted. Each section is named to match a TI process in the server.

Each TI process section is used to define the parameters of the TI process and their default values.

If there are multiple TI process sections with the same name, only the first one will be used.

## 2.4.4 Example config file

The example below shows the [RunTI] section and a section for a single TI process ("my\_ti\_process"). The parameters and their default values (which may be overridden by parameters provided on the command line) are defined below each section header.

```
[TM1RunTI]
process=my_ti_process
connect=Production

[Process - my_ti_process]
num1="value1"
stringX="value2"
stringY="value3"

[Connect - Production]
adminhost=
server=MyTM1server
user="MyTM1AdminServer"
pwdfile="c:\tml_admin_area\passwords\tml_password.txt"
AdminSvrSSLCertAuthority=.\ssl\applxca.pem
AdminSvrSSLCertID=tmladminserver
AdminSvrSSLCertRevList=
CAMNamespace=LOCAL_NTLM
```

## 2.4.5 Processing logic

Configuration parameters and command line parameters are processed in the following fashion:

1. If specified by -i, the configuration file is opened and any connect option specified in [TM1RunTI] is processed first.

2. Any other parameters in [TM1RunTI] are then processed (and may override those specified by the connect parameter)
3. The command line parameter -connect is processed next, if present. It will load values from the associated [Connect - <connection\_name>] section of the config file, overriding any values loaded by the preceding steps.
4. The remaining command line parameters are processed.

For example, let's say the config file in the preceding example is saved with the name tm1tools.config and then the user executes the following:

```
tmlrunTI -i ".\tm1tools.config" -passwordkeyfile
c:\keystore\prodkey.dat -connect prodsystem
```

Since the -i parameter was provided, the tool would do the following:

1. open the config file and load the [tm1guardian] section
2. upon seeing the connect parameter in [tm1guardian], load the parameter values from [Connect - testsystem]
3. process the command line parameters:
  - a. upon seeing the connect parameter, load the parameters from [Connect – prodsystem]
  - b. replace the value for passwordkeyfile.

### 2.4.6 Configuration filename and location

The TM1RunTI command line parameter “-i” can be used to specify a configuration filename. This is particularly useful if several TM1 servers are to be supported in the environment, as a different configuration file can be used for each server and like-named TI’s in different servers can be defined with different parameters.

## 2.5 Return Codes and Error Messages

Return Code	Message	Description
0	None	The program completed successfully.
1	Short Help text	No parameters were provided. The short help is sent to stderr. Equivalent to -?
	Invalid number of parameters at <n>	More parameters were detected than are actually supported by the program, beginning at the n'th parameter.
2	Server connection failed	The program was unable to make a connection to a TM1 server.

Return Code	Message	Description
3	Calling process: <TI_name> completed with minor errors.	The TI process completed but with minor errors.
4	Calling process: <TI_name> completed with messages	The TI process completed but returned messages.
5	<p>Error retrieving password</p> <ul style="list-style-type: none"> <li>- NULL key returned from reading &lt;filename&gt; key file</li> <li>- NULL password returned from reading &lt;filename&gt; password file</li> <li>- Error obtaining file status of &lt;filename&gt;</li> <li>- Error opening &lt;filename&gt;</li> <li>- Unable to allocate data for key</li> <li>- Error reading &lt;filename&gt; key file</li> </ul>	<p>The program was unable to get the password from the password file.</p> <p>One of the other error messages listed may appear in stderr before this one, indicating more precisely the nature of the problem.</p>
99	Other TI error	The TI process completed with an unspecified error.

## 3. Other Considerations

### 3.1.1 Authentication

During the login process, RunTI determines which security mode the server is in and chooses the appropriate method of authentication to connect to the server. The following modes are supported:

- Native TM1 security
- LDAP Authentication
- Cognos Access Manager (CAM)
- Integrated Security

### 3.1.2 Password Security

The use of passwords on the command line for `tm1runti` is not recommended for production deployments. Rather, the password should be passed to the program by using the *passwordfile* parameter to specifying a file that contains the encrypted password. A keyfile is also needed, to decrypt the password, and this is provided via the *passwordkeyfile* parameter. These files can be stored in a location accessible to the username running TM1RunTI, but under operating system protection so that other users cannot access them.

A combination of password and key can be generated by using TM1Crypt tool which comes with the standard TM1 installation. See the TM1 Operations Guide for details.

### 3.1.3 Modes of Execution and Error Handling Limitations

TM1RunTI can be run as a standalone executable, from within an operating system batch script, or from within a TM1 TurboIntegrator process.

The most straightforward way to run TM1RunTI from within TurboIntegrator is to use the `ExecuteCommand()` call to directly execute it. For example:

```
ExecuteCommand("tm1runti -i myconfig.config -connect prodserver -name update1")
```

The ability to define connection and other relatively static parameters in a configuration file makes it possible to simplify the parameter list passed to TM1RunTI from a calling TI, and to reduce maintenance effort by centralizing connection information.

Executing TM1RunTI directly from within a TI using `ExecuteCommand()` has an important limitation. TM1RunTI will return an error code if it fails, but the `ExecuteCommand()` does not return the error code and there is no other mechanism in TurboIntegrator to access the return code after the call.

To deal with errors, it is advisable that TM1RunTI be executed instead from a batch script (called by `ExecuteCommand()`) so that the error return code can be obtained (in `CMD.EXE` via the `ERRORLEVEL` variable) and so that error messages can be logged or intercepted by redirecting `stderr`. Various options are then available to the application designer for handling the error, such as:

- Writing the error information to database.

- Writing the error information to a file and then, in a subsequent TI, loading the information into a TM1 Cube. The cube can then be used for reporting, alerts, etc. (Caveat: in TM1 versions 9.5.1 and earlier, this could create additional lock contentions).
- Writing the error information to a file (or files) and then, within the calling TI, use the FileExists() TI function to test for the existence of that file (or files). The TI can then take conditional actions based on the existence of the file(s) generated by the batch script.

## Appendix A: Definitions, Acronyms and Abbreviations

Term	Definition
BLM	Bulk Load Mode. BLM is a mode where one thread is granted exclusive access to the server. If a call to thread X is made to switch on BLM, the server waits until all in-process operations are finished and then suspends all sessions/threads except X. Thread X is then allowed to proceed with exclusive control of the server. While the server is in BLM (or about to enter BLM), all new attempts to connect to the server are refused.
CAM	Cognos Access Manager. A Cognos platform component that provides a common API for authentication to various supported authentication systems such as LDAP, Active Directory, and others.
CLI	Command Line Interface – a program that runs in a command shell (such as Windows cmd.exe or Unix c-shell) and which may accept various parameters and values on the command line to specify required information or to control optional behavior. A CLI generally uses stdin for input and stdout for output, and stderr for error messages, but it may also interoperate with other processes to update data, send messages, etc.
Console App	Windows synonym for CLI.
Filespec	A File Specification. On Windows, a fully qualified filename, a partially qualified filename, or an unqualified filename.

Table 1 – Definitions, Acronyms and Abbreviations

## Appendix B: References

- TM1 Operations Guide

## Appendix C: Help Text

### NAME

tmlrunti.exe - Run a specified TM1 TurboIntegrator (TI) process

### SYNOPSIS

```
tmlrunti -?
or tmlrunti -help
or tmlruntl [<cmd_parm>...] [<ti_parm>...]
```

where <cmd\_parm> is one of:

```
-i <filespec>
-process <string>
-connect <string>
<connect_parm>...
```

where <ti\_parm> is:

```
<parm_name> '=' <parm_value>
```

where <connect\_parm> is one of:

```
-adminhost <string>
-server <string>
-user <string>
<password_parm>
-AdminSvrSSLCertAuthority <filespec>
-AdminSvrSSLCertID <id>
-AdminSvrSSLCertRevList <filespec>
-AdminSvrSSExportKeyId <id>
-ExportAdminSvrSSLCert <T|F>
-CAMNamespace <string>
```

where <password\_parm> is one of:

```
-pwd <string>
-passwordfile <filespec> -passwordkeyfile <filespec>
```

### DESCRIPTION

tmlrunti.exe is a command line program which can execute a specified TurboImage (TI) process on a TM1 server.

To reduce the number of parameters that need to be provided when executing tmlrunti, parameters with relatively constant values can be provided in a configuration file.

The program accepts two kinds of parameters: command options that specify things the program needs for execution; and TI parameters, which are passed through to the TI being executed.

### COMMAND PARAMETERS

```
-i <filespec>
  Filename of the file containing default configuration parameters.
  Parameters specified in this file will be used, unless overridden by
```

parameters provided on the command line. If no path is specified, the TM1 server directory will be assumed. If -i is not specified, then other parameters must be specified in order to provide the process name, TM1 server, etc.

- connect <string>  
This parameter can be used to specify a section in the config file containing parameters used to make server connections, such as user, pwd, CAMnamespace, etc.
- process <string>  
Name of the TI process that is to be executed.
- help  
Long help. Display help documentation including parameters and descriptions.
- ? Short help. Display a synopsis of command line parameters.

CONNECT PARAMETERS

- adminhost <string>  
TM1 admin hostname.
- server <string>  
TM1 server name.
- user <string>  
TM1 or Cognos Access Manager (CAM) username, depending on the type of authentication being used by the TM1 server.
- AdminSvrSSLCertAuthority <filespec>  
The full path of the certificate authority file that issued the TM1 Admin Server's certificate.
- AdminSvrSSLCertID <id>  
The name of the principal to whom the TM1 Admin Server's certificate is issued. Note: The value of this parameter should be identical to the SSLCertificateID parameter in the tmladmsrv.ini file.
- AdminSvrSSLCertRevList <filespec>  
The full path of the certificate revocation file issued by the certificate authority that originally issued the TM1 Admin Server's certificate. A certificate revocation file will only exist in the event a certificate had been revoked.
- ExportAdminSvrSSLCert <T|F>  
Specifies whether you want the certificate authority certificate which originally issued the TM1 Admin Server's certificate to be exported from the Windows certificate store at runtime. When this option is selected, you must also set a value for AdminSvrSSLExportKeyId as described below.
- AdminSvrSSLExportKeyId <id>  
The identity key used to export the certificate authority

certificate, which originally issued the TM1 Admin Server's certificate, from the certificate store. This parameter is required only if you choose to use the certificate store by setting `ExportAdminSvrSSLCert=T`.

`-CAMNamespace <id>`

Id of the Cognos Access Manager (CAM) namespace. Note, this is the namespace id, not the namespace name.

#### TI PARAMETERS

`<parm_name>=<parm_value>`

The TurboIntegrator (TI) process specified by `-process` may be capable of taking one or more parameters values. If needed, they must be specified after all command parameters. Each `<parm_name>` is a string matching a parameter name defined for the TI. Each `<parm_value>` is a character string or number string, depending on the value expected by the TI. Example: `MyTIparm="my_value"`

#### PASSWORD PARAMETERS

`-pwd <string>`

Password for the username given in the `-user` parameter, in clear text. For greater security, the password can be specified in an encrypted file using the `-passwordfile` parameter.

`-passwordfile <filespec>`

Filename of the file containing the encrypted password for the user specified by `-user`. If no path is specified, the TM1 server directory will be assumed. When this option is used, you cannot use `-pwd`.

`-passwordkeyfile <filespec>`

If the `passwordfile` parameter is given, a key file is also required in order to decrypt the password. The password file and key file can be created using TM1Crypt tool. (See the TM1 Operations Guide.)

#### COPYRIGHT AND LICENSE

Copyright 2011, IBM Corporation