



# *TrueNet FX*<sup>TM</sup> API Guide

System Version 6.04.06

# **TrueNet FX API™**

## **Software License Agreement**

This is a legal agreement between you and R-Quest Technologies, LLC. ("R-Quest"), covering your use of TrueNet FX API (the "Software"). Be sure to read the following agreement before using the Software. BY USING THE SOFTWARE (REGARDLESS IF YOU HAVE REGISTERED THE SOFTWARE OR NOT), YOU ARE AGREEING TO BE BOUND BY THE TERMS OF THIS AGREEMENT. IF YOU DO NOT AGREE TO THE TERMS OF THIS AGREEMENT, DO NOT USE THE SOFTWARE AND DESTROY ALL COPIES IN YOUR POSSESSION.

The Software is owned by R-Quest and is protected by United States copyright laws and international treaty provisions. Therefore, you must treat the Software like any other copyrighted material (e.g., a book or musical recording). Paying the license fee allows you the right to use one copy of the Software on a single computer.

You may not use this software to copy copyright material without the permission of the copyright owner. You may not rent or lease the Software, nor may you modify, adapt, translate, reverse engineer, decompile, or disassemble the Software. If you violate any part of this agreement, your right to use this Software terminates automatically and you must then destroy all copies of the Software in your possession.

The Software and its related documentation are provided "AS IS" and without warranty of any kind and R-Quest expressly disclaims all other warranties, expressed or implied, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. Under no circumstances shall R-Quest be liable for any incidental, special, or consequential damages that result from the use or inability to use the Software or related documentation, even if R-Quest has been advised of the possibility of such damages. In no event shall R-Quest liability exceed the license fee paid, if any.

Any written or oral information or advice given by *R-Quest* dealers, distributors, agents or employees will in no way increase the scope of this license or warranty. Nor may you rely on such oral or written communication. Some states do not allow the exclusion or limitation of implied warranties or liability for incidental or consequential damages, so the above limitation or exclusion may not apply to you. This warranty gives you specific legal rights, and you may also have other rights which vary from state to state.

This Agreement shall be governed by the laws of the State of California. If for any reason a court of competent jurisdiction finds any provision of the Agreement, or portion thereof, to be unenforceable, that provision of the Agreement shall be enforced to the maximum extent permissible so as to effect the intent of the parties, and the remainder of this Agreement shall continue in full force and effect.

## **COPYRIGHT**

Copyright © 2002-2009 R-Quest Technologies, LLC.

**All rights reserved.** This manual and the software described in it are copyrighted with all rights reserved. No part of this publication may be reproduced, transmitted, transcribed, stored in a retrieval system or translated into any language in any form by any means without the written permission of R-Quest.

## **TRADEMARKS**

**“R-Quest”, “TrueNet”, “TrueNet FX”, TrueNet API”, “TrueNet FX API”, “TrueISO”, “TrueCopy”, “TrueCopy System Technology” and “Powered By R-Quest”** are trademarks of **R-Quest Technologies, LLC**. Other brand names and product names are trademarks or registered trademarks of their respective companies. **R-Quest Technologies, LLC** may also claim other names, words or phrases contained in this manual as trademarks.

## **Materials Warranty**

If you discover physical defects in the media on which *R-Quest* software is distributed, or in the User Manual supplied by *R-Quest*, *R-Quest* will replace the media or manual for a period of ninety (90) days after the purchase of the retail customer. You must return the media or manual to *R-Quest* or an authorized *R-Quest* dealer within the warranty period accompanied by proof of purchase.

## **Piracy / Illegal Copying**

TrueNet FX API™ is designed to help you easily reproduce material for which you own the copyright or have obtained permission from the copyright owner to make copies. Unless you own the copyright or have permission from the copyright owner to make copies, you are violating the software license agreement, and may be violating copyright law and be subject to payment of fines, damages and / or other remedies. *R-Quest* accepts no responsibility for the use or misuse of this equipment for illegal purposes. If you are uncertain about your rights, contact your legal advisor.

# TrueNet FX API

<b>Setting up the TrueNet FX API .....</b>	<b>7</b>
<i>Installing TrueNet FX API .....</i>	<i>7</i>
<i>Running TrueNet FX API.....</i>	<i>7</i>
<i>Defining Network Devices.....</i>	<i>7</i>
<b>Introduction to the API.....</b>	<b>10</b>
<b>Guide to API file extensions.....</b>	<b>11</b>
<b>Keywords .....</b>	<b>12</b>
<b>ABSTRACT .....</b>	<b>12</b>
<b>AUTO_DELETE .....</b>	<b>12</b>
<b>BUILD_PATH .....</b>	<b>12</b>
<b>BUILD_TYPE .....</b>	<b>13</b>
<b>COPYRIGHT.....</b>	<b>13</b>
<b>DEVICE .....</b>	<b>13</b>
<b>FAST_START.....</b>	<b>14</b>
<b>FIXATE .....</b>	<b>14</b>
<b>IMAGE_FILE .....</b>	<b>14</b>
<b>IMAGE_TYPE.....</b>	<b>15</b>
<b>JOB_NAME.....</b>	<b>16</b>
<b>JOB_TYPE .....</b>	<b>16</b>
<b>JOLIET .....</b>	<b>16</b>
<b>MERGE_SOURCE_# (where # is a decimal number).....</b>	<b>17</b>
<b>MERGE_TARGET_# (where # is a decimal number).....</b>	<b>17</b>
<b>PREPARER .....</b>	<b>17</b>
<b>PRINT_BACKGROUND.....</b>	<b>18</b>
<b>PRINT_FILE_ENTRY_# (where # is a decimal number).....</b>	<b>18</b>
<b>PRINT_QUALITY .....</b>	<b>18</b>
<b>PRINT_MEDIA .....</b>	<b>19</b>
<b>PRINT_TEMPLATE.....</b>	<b>19</b>
<b>PUBLISHER.....</b>	<b>20</b>
<b>QUANTITY.....</b>	<b>20</b>
<b>SELECT_MEDIA .....</b>	<b>20</b>
<b>TRACK_# (where # is a decimal number).....</b>	<b>21</b>
<b>VOLUME.....</b>	<b>21</b>

<i>WRITE_SPEED</i> .....	22
<b>Additional Status Added To Command Files By TrueNet FX API</b> ....	23
<i>ACCEPT</i> .....	23
<i>REJECT</i> .....	23
<i>STATUS</i> .....	23
<i>JOB_HANDLE</i> .....	23
<b>Remote Device Status Files (.rds)</b> .....	24
<i>DEVICE_NAME</i> .....	25
<i>DEVICE_IP</i> .....	25
<i>FIRMWARE</i> .....	25
<i>ONLINE</i> .....	25
<i>JOBS_QUEUED</i> .....	25
<i>JOBS_RUNNING</i> .....	25
<i>JOBS_DONE</i> .....	25
<i>HOPPER_COUNT</i> .....	25
<i>HOPPER_#_MEDIA</i> .....	26
<i>HOPPER_#_TYPE</i> .....	26
<i>HOPPER_#_COUNT</i> .....	26
<i>RECORDERS</i> .....	26
<i>CD_RECORDERS</i> .....	26
<i>DVD_RECORDERS</i> .....	26
<i>BD_RECORDERS</i> .....	26
<i>BD_DL_RECORDERS</i> .....	26
<i>RECORDER_#_ACTIVITY</i> .....	26
<i>RECORDER_#_PROGRESS</i> .....	26
<i>HDD_SPACE</i> .....	27
<i>PRINTER</i> .....	27
<i>PRINTER_CONSUMABLES</i> .....	27
<i>SYSTEM_STATUS</i> .....	27
<b>Appendix A</b> .....	29
<i>Building / Merging ISO / UDF File Systems</i> .....	29
<b>Appendix B</b> .....	32
<i>Example Command Files</i> .....	32
<b>Appendix C</b> .....	36
<i>API Command File Error Codes</i> .....	36
<b>Appendix D</b> .....	39
<i>API Job Status Codes</i> .....	39

**Appendix E.....42**  
*Running from the Command Line.....42*  
*Headless Operation.....42*  
*Manually configuring the API devices .....43*

# Setting up the TrueNet FX API

The TrueNet FX API launcher is installed on your computer (or server) and runs as a native application (and not via a web browser). This has the advantage of giving much greater control over the Java environment than is available within a Web Browser environment.

## Installing TrueNet FX API

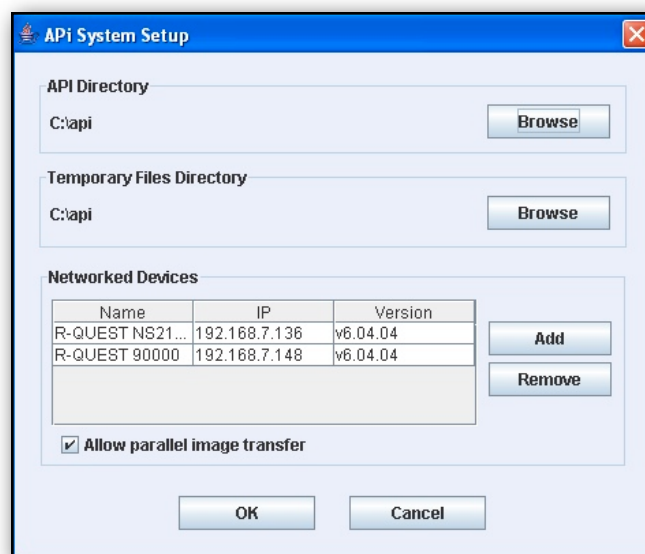
To install the TrueNet FX API application, double click on the installer icon and follow the on screen prompts. Once the application is installed, double click the application to run it. Follow the on-screen prompts.

## Running TrueNet FX API

When running TrueNet FXAPI for the first time you need to enter the IP address (or URL) of the primary system on your network. If you only have one system, enter the IP address for that system. Once the IP address (or URL) has been entered, click the RUN button. The TrueNet FX API launcher will now retrieve the latest software version from the primary networked system before displaying the API user interface. The FX API software is automatically updated as part of system firmware so that you are always running matching software and system firmware on the primary system. You should always update all systems on the network to the same version when using them with TrueNet FX API.

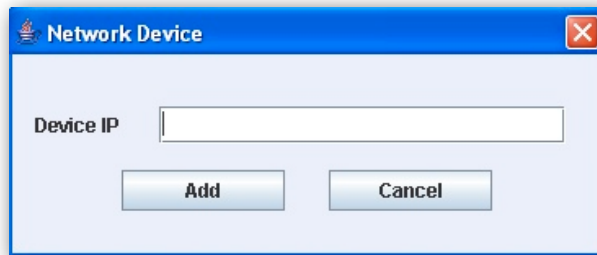
## Defining Network Devices

When first launched there will be no network devices defined. Select the Setup menu, then Network Setup. A dialog will appear that allows you to set where the API files can be found, and what network devices to use.



Click the top right Browse button to browse for the directory where the API files will be dropped (this directory should already exist), and the second Browse button to browse for a directory that TrueNet FX API can use for any temporary files. This can be the same as the API directory.

Once the API and Temporary directories have been defined click the “Add” button to add a new device to the API system. The following dialog will appear:



Enter the IP address or URL for the network system, then click the ‘Add’ button. To exit without creating a new device click the ‘Cancel’ button.

TrueNet FX API will attempt to connect to the new system and obtain information about it, including the designated system name. This is the name that all API files should use when sending jobs to it. To change the system name, log on to the system (as an administrator) using a Web Browser, then in the Administration dialog, Network Tab, enter a new name for the system. Note that after changing a system name, TrueNet FX API should be restarted.

Once all the devices have been added, click the ‘OK’ button to exit the API System Setup dialog. A new tab should appear for each of the devices showing the hopper types, installed recorders (with near-real-time activity indicators), and printer consumables levels. Also shown are the current API jobs queued / running on the system, and the status of any image transfers that are taking place. Each tab represents one system.

The devices used by the system are stored in a configuration file. Normally you will not need to edit this file, but if for any reason you want to manually edit devices, look for the file *TrueNetFX\_Settings.fxs* in your user directory. This file will contain entries for each device, e.g.:

```
DEVICE_COUNT=2
DEVICE_IP_0=192.168.7.136
DEVICE_IP_1=192.168.7.137
```

The DEVICE\_COUNT line specifies how many active devices there should be, and each device will have its IP address (or URL) listed. If you manually add or remove a device in this file, ensure that the DEVICE\_COUNT is correct, and that the devices are listed, and numbered sequentially starting at 0. If any number is skipped all devices that follow will be missing.

## TrueNet FX API

The purpose of the TrueNet FX API application is to run API jobs, and is not intended as a Device administration platform. Because TrueNet FX API provides very limited user interactivity with the Devices itself, all system administration and other manual job manipulation should be done by using the Web Browser interface.

# Introduction to the API

TrueNet FX API is an **A**pplication **P**rogramming **I**nterface that allows custom job requests to be sent to the TrueNet FX API software, including copying, comparing, custom data, video and audio mastering, together with custom disc printing.

API command files (think of these as “job scripts” or “order files”) are plain ASCII text files that are created by a user-provided program, or created by a user with any plain ASCII text editor, such as Windows Notepad.

Command files contain information about the task required, and are in the form of IDENTIFIER=PARAMETER. Command files are placed in a predetermined sub directory (configure this via the **Setup | API System** menu) that is scanned approximately every 5 seconds by TrueNet FX API.

To prevent TrueNet FX API starting a job using a file that is not yet completed (i.e. still being written by your application) the file should initially be created without a ‘.job’ file extension (or any of the other reserves file extensions - see below), and renamed with a ‘.job’ extension only once the file is complete.

When a new “.job” file is detected by TrueNet FX API, the file is read, and if there are no errors, the file is renamed with a file extension of “.bsy” (busy) and the job is added to the Job Queue. If errors are encountered when parsing the file it is renamed with a “.bad” extension. Syntax errors or other errors such as missing image files are reported in the command file. Parsing will halt at the first error.

If a job should be run at a higher priority than other jobs, create the job with a “.hi” extension instead of “.job”. That will be the next job to scanned and sent to the system for processing, and once on the system will also take priority over other (none high priority) jobs.

When the job is completed, the file is renamed with the extension “.don” (done) (or automatically deleted, depending on the state of the “AUTO\_DELETE” setting).

If jobs encounter an error while running, for example if the input hoppers become empty, the job will be temporarily changed to “.err”. An error code and text description will indicate the nature of the problem. When the problem has been corrected (usually by an operator attending the system) the job will resume and automatically be renamed to “.bsy” again. You should not normally need to rename “.err” files manually.

The file can be monitored by the user application, and the status assessed via the following keywords: ACCEPTED (shows the number of good discs so far), REJECTED (shows the number of discs rejected), and STATUS.

API files allow for a very flexible solution-based approach to disc production and/or printing. Customised mastering (e.g. Unique files / serialisation) is both possible and practical to do, as is customised printing. Custom printing allows each disc to be printed

with different text and/or graphics simply by changing the commands sent in each file.

By combining the custom build and custom print options, every disc can be unique and tailored to your exact requirements.

Lines beginning with # are comments, and are ignored by the file parser.

## Guide to API file extensions

Extension	Informative
.job	Used for a job file added to the API queue. This file is created by the user application.
.hi	The same as a .job file, but uses a “.hi” extension to indicate it’s a high priority job. All files with a .hi extension will be parsed before files with a .job extension. When a high priority job is sent to the system it will automatically be elevated above other ‘Normal’ priority jobs in the queue, and run at a higher priority.
.bsy	A file that has already been parsed by Truenet FX API and is currently being processed is renamed (by Truenet FX API) to a .bsy (busy) file.
.err	<p>The job is currently in an error state and is paused. The last error is indicated by the ERROR_CODE in the file. Most errors are <b>recoverable by simple operator actions</b>, and exist due to events such as ‘Input Hopper Empty’. In this case once the input hopper has been refilled by the operator the job will return to a .bsy state automatically. Most errors are <b>not fatal</b>.</p> <p>A job with a “.err” extension that also has an ERROR_CODE=0 means that there is a system wide error (such as input hopper empty) rather than a job specific error. See Appendix C for error codes and Appendix D for Status codes.</p>
.bad	Errors were encountered while parsing the .job (or .hi) file. Check the file for the error number and description. See Appendix C for error codes.
.don	The job is complete. Whenever a running job completes successfully the file is renamed with a .don extension.

# Keywords

In the following section the keywords are listed in alphabetical order, with a list of prerequisite keywords (i.e. keywords that must be present, and used correctly before this keyword has any effect), together with hints on when to use and when not to use this keyword. It is important to understand that not all keywords will be used in a command file, and some keywords depend on the presence of others.

## ABSTRACT

**Prerequisite :** DEVICE, JOB\_TYPE (=BUILD), BUILD\_TYPE, BUILD\_PATH

**When to use :** When building an ISO or UDF file system disc from scratch.

**When *not* to use :** When using a disc image that has already been created, or print only jobs.

**See Examples :**

Description:

ABSTRACT is a text element of an ISO9660 volume descriptor, and is a place holder for abstract information. This field is optional, but when used has a maximum 128 characters. Most authors do not use this field.

---

## AUTO\_DELETE

**Prerequisite :** DEVICE, JOB\_TYPE

**When to use :** When you don't want to have to delete the API command file yourself, and do not need to see the results of a job.

**When *not* to use :** When ever you need to know if a job completed successfully.

**See Examples :**

Description:

This parameter is optional, but when set to YES, TrueNet FX API will **delete** the job file on completion instead of renaming the job file to ".don".

---

## BUILD\_PATH

**Prerequisite :** DEVICE, JOB\_TYPE (=BUILD), BUILD\_TYPE

**When to use :** When building an ISO or UDF file system disc from scratch.

**When *not* to use :** When building an AUDIO\_CD or when using a disc image that has already been created, or when creating print only jobs.

**Also See :** BUILD\_TYPE

**See Examples :** 3, 6, 10

Description:

When BUILD\_TYPE=ISO\_CD, ISO\_DVD, UDF\_DVD or UDF\_BD, this keyword specifies the source parent directory where the files are located. TrueNet FX API will not

create a physical image file on your local hard disk, but will build the image as the files are moved across the network to your duplicator. This saves the time to build the file, and also local hard disk space.

---

## **BUILD\_TYPE**

**Prerequisite** : DEVICE, JOB\_TYPE (=BUILD), BUILD\_PATH

**When to use** : When building an ISO or UDF file system disc from scratch.

**When *not* to use** : When using a disc image that has already been created, or print only jobs.

**Also See** : BUILD\_PATH, JOLIET

**See Examples** : 3,6,10,11

Possible settings are (settings are mutually exclusive):

BUILD\_TYPE=ISO\_CD (**Also See** : JOLIET)

BUILD\_TYPE=ISO\_DVD

BUILD\_TYPE=UDF\_DVD

BUILD\_TYPE=UDF\_BD

BUILD\_TYPE=AUDIO\_CD

The BUILD\_TYPE keyword must be present when ever the BUILD parameter is used with JOB\_TYPE, and also requires the BUILD\_PATH to point to the source path for the data for all types except AUDIO\_CD. For AUDIO\_CD see TRACK\_#.

---

## **COPYRIGHT**

**Prerequisite** : DEVICE, JOB\_TYPE (=BUILD), BUILD\_TYPE, BUILD\_PATH

**When to use** : When building an ISO or UDF file system disc from scratch.

**When *not* to use** : When using a disc image that has already been created, or print only jobs.

**See Examples** :

Description:

COPYRIGHT is a text element of an ISO9660 volume descriptor, and is a place holder for copyright information. This field is optional, but when used has a maximum 128 characters. Most authors do not use this.

---

## **DEVICE**

**Prerequisite** : JOB\_TYPE

**When to use** : Every API job requires this keyword.

**When *not* to use** :

**See Examples** : All examples

Description:

DEVICE specifies the target network device name (not IP address). This is the same name that was given to the network device as part of the network device setup. Do not include the IP address in this name. An example is: "DEVICE=R-QUEST NS2100" where there is a device with the device name of "R-QUEST NS2100" setup in the API software.

---

## **FAST\_START**

**Prerequisite :** DEVICE, JOB\_TYPE (=BUILD or COPY)

**When to use :** When using a pre-existing local disc image or building an ISO or UDF file system disc from scratch.

**When *not* to use :** When the disc image already exists on the network system hard disk, or where the network connectivity is slow or unreliable.

**Default :** NO (if not present FAST\_START is not used)

**See Example :** 9

Description:

The fast start option allows the system to begin loading the disc and start recording as soon as the data begins arriving from the API host. This saves time by overlapping the data transfer with the recording process.

This is an optional parameter, and if not present defaults to NO. Possible settings are YES or NO.

---

## **FIXATE**

**Prerequisite :** DEVICE, JOB\_TYPE (=BUILD or COPY)

**When to use :** When using a pre-existing local ISO (not RQI) disc image or building an ISO or UDF file system disc from scratch.

**When *not* to use :** When the disc image already exists on the network system hard disk, or the image type is '.rqi'.

**Default :** YES

**See Examples :** 9,10

TrueNet FX API can produce CDs that are either fixated (additional sessions cannot be added to the written disc) or not fixated (additional sessions can be added to the written disc). This is optional, and when not present defaults to FIXTATE=YES.

To allow additional sessions to be added to copied discs, use FIXATE=NO.

---

## **IMAGE\_FILE**

**Prerequisite :** DEVICE, JOB\_TYPE (=COPY or COMPARE), IMAGE\_TYPE

**When to use :** When using a pre-existing local ISO or RQI disc image

**When *not* to use :** When the disc image already exists on the network system hard

disk, or when building a new disc image (using JOB\_TYPE=BUILD).

Also See : IMAGE\_TYPE

**See Examples :** 1,2,8

**Description:**

Full path & name of the image file. Where the COPY parameter is used with the JOB\_TYPE, and BUILD is NOT used, then TrueNet FX API needs a completed image file. This can be one of several different types (see IMAGE\_TYPE). The parameter is the full path and file name of the image file, e.g. IMAGE\_FILE=c:\images\image.rqi

When building an image using JOB\_TYPE=BUILD this field is **ignored**.

---

**IMAGE\_TYPE**

**Prerequisite :** DEVICE, JOB\_TYPE (=COPY or COMPARE), IMAGE\_FILE

**When to use :** When using a pre-existing local ISO or RQI disc image

**When not to use :** When the disc image already exists on the network system hard disk, or when building a new disc image (using JOB\_TYPE=BUILD).

Also See : IMAGE\_FILE

**See Examples :** 1,2,8

**Description:**

There are three types of image file that can be used by TrueNet FX API. They are:

RQI - The native R-Quest Image File

RAW\_CD – a complete binary image of a mode 1 CD. All blocks must be 2048 bytes. An ISO file is typically a 2048 byte binary image.

RAW\_DVD - a complete binary image of a DVD (2048 byte blocks), e.g. an .ISO file.

RAW\_BD - a complete binary image of a BD(2048 byte blocks), e.g. an .ISO file (requires a Blu-ray licensed system with installed Blu-ray recorders).

RQI files can contain any type of discs that are supported by the duplication system, including Data, Audio, Video, Mixed mode, Multi session etc.

Examples:

IMAGE\_TYPE=RQI

IMAGE\_TYPE=RAW\_CD

IMAGE\_TYPE=RAW\_DVD

IMAGE\_TYPE=RAW\_BD

This field is mandatory when using BUILD\_TYPE=COPY with an IMAGE\_FILE, but ignored when using the BUILD command.

## **JOB\_NAME**

**Prerequisite :** DEVICE

**When to use :** Optional for all API jobs files

**When *not* to use :** Use is entirely optional

**See Examples :**

### **Description:**

JOB\_NAME allows the user to specify the text that appears on the API GUI next to a job. If this is not present, the name of the API file will be used instead.

---

## **JOB\_TYPE**

**Prerequisite :** DEVICE

**When to use :** All API job files

**When *not* to use :** Never

**See Examples :** ALL

### **Description:**

This is the type of job required. There are four types of job; COPY, COMPARE, BUILD, PRINT. AUDIT is option when used in combination with the COPY command, but cannot be used on it's own.

COPY is implied by the BUILD command. There are no requirements for spaces or other delimiters, but the preferred method for ease or reading by humans is to add a '+' between the words. An example of Copy and Print would be:

```
JOB_TYPE=COPY+PRINT
```

---

## **JOLIET**

**Prerequisite :** DEVICE, JOB\_TYPE (=BUILD)

**When to use :** When building an ISO image from scratch

**When *not* to use :** Any type of job other than an ISO build.

Also See : JOB\_TYPE, BUILD\_TYPE

**See Examples :**

### **Description**

JOLIET acts as a modifier to ISO\_CD and ISO\_DVD when set via BUILD\_TYPE. If this is not present, the system defaults to JOLIET=NO. JOLIET=YES will add a Joliet file

structure to an ISO9660 master. Joliet provides a relaxed structure for long files names and other directory naming conventions, together with Unicode directory information.

---

**MERGE\_SOURCE\_# (where # is a decimal number)**

**Prerequisite :** DEVICE, JOB\_TYPE (=BUILD), MERGE\_TARGET\_#

**When to use :** When building an ISO image from scratch and you want to merge a file from a location other than the main data source.

**When not to use :** Any type of job other than an ISO build.

Also See : JOB\_TYPE, BUILD\_TYPE, MERGE\_TARGET\_#

**See Example : 6**

**Description:**

When BUILD\_TYPE=ISO\_CD or ISO\_DVD, a Merge Source specifies an optional merge source path for files to be merged when the ISO image is built. This is ideal for use in creating unique discs, when only a small number of files change. Create the main part of the disc from a static directory structure and merge only the changed files. See the notes on Building/Merging ISO.

---

**MERGE\_TARGET\_# (where # is a decimal number)**

**Prerequisite :** DEVICE, JOB\_TYPE (=BUILD), MERGE\_SOURCE\_#

**When to use :** When building an ISO image from scratch and you want to merge a file from a location other than the main data source.

**When not to use :** Any type of job other than an ISO build.

Also See : JOB\_TYPE, BUILD\_TYPE, MERGE\_SOURCE\_#

**See Example : 6**

**Description:**

Used to specify where files from a Merge Source will be merged in to on the target ISO file system. See the notes on Building/Merging ISO.

---

**PREPARER**

**Prerequisite :** DEVICE, JOB\_TYPE (=BUILD), BUILD\_TYPE, BUILD\_PATH

**When to use :** When building an ISO or UDF file system disc from scratch.

**When not to use :** When using a disc image that has already been created, or print only jobs.

**See Examples :**

**Description:**

Preparer is a text element of an ISO9660 volume descriptor, and is a place holder for preparer information. This field is optional, but when used has a maximum 128 characters.

## **PRINT\_BACKGROUND**

**Prerequisite :** DEVICE, JOB\_TYPE (=PRINT), PRINT\_TEMPLATE

**When to use :** When you need to change the background image when printing from a template file

**When *not* to use :** When using a print image that has already been created (i.e. a PRN file) or the job does not include printing.

**See Example :** 7

Description:

When using a print template, a JPG (jpeg) file can be specified as a replacement background. The JPG file will be automatically scaled to fit the disc size using the shortest side.

Example:

PRINT\_BACKGROUND=c:\images\print.jpg

This is optional, and only used with PRINT\_TEMPLATE.

---

## **PRINT\_FILE\_ENTRY\_# (where # is a decimal number)**

**Prerequisite :** DEVICE, JOB\_TYPE (=PRINT), PRINT\_TEMPLATE

**When to use :** When you need to specify dynamic text for a pre-defined text entry on a template file.

**When *not* to use :** When using a print image that has already been created (i.e. a PRN file) or the job does not include printing.

**See Example :** 6, 7, 8

Description:

Using a print template with embedded 'dynamic tags' allows TrueNet FX API to produce dynamic (or unique) print files. # = a number starting at 1 and incrementing by 1 for each field. Where a text string in a template file has "%#%" included (where #=a decimal number) the corresponding PRINT\_ENTRY\_# supplies the substitute text. These are completely optional, and are only used with PRINT\_TEMPLATE. Example:

PRINT\_ENTRY\_1=First text to replace

PRINT\_ENTRY\_2=This is the second line

---

## **PRINT\_QUALITY**

**Prerequisite :** DEVICE, JOB\_TYPE (=PRINT), PRINT\_TEMPLATE

**When to use :** When you need to programatically control the quality of the printed output, and should be used in combination with PRINT\_MEDIA. These are used only

with InkJet printers such as the R-Quest FlashJet 4800, FlashJet 2 and FlashJet Pro.

**When not to use** : When using a print image that has already been created (i.e. a PRN file) or the job does not include printing, or with printers other than the FlashJet series listed above.

See also : PRINT\_MEDIA

Possible Settings:

PRINT\_QUALITY=TEXT

PRINT\_QUALITY=FAST

PRINT\_QUALITY=NORMAL

PRINT\_QUALITY=BEST (default)

---

## PRINT\_MEDIA

**Prerequisite** : DEVICE, JOB\_TYPE (=PRINT), PRINT\_TEMPLATE

**When to use** : When you need to programatically control the quality of the printed output, and should be used in combination with PRINT\_QUALITY. These are used only with InkJet printers such as the R-Quest FlashJet 4800, FlashJet 2 and FlashJet Pro.

**When not to use** : When using a print image that has already been created (i.e. a PRN file) or the job does not include printing, or with printers other than the FlashJet series listed above.

See also : PRINT\_QUALITY

Possible settings are:

PRINT\_MEDIA=PLAIN

PRINT\_MEDIA=MATTE

PRINT\_MEDIA=PREMIUM

PRINT\_MEDIA=PHOTO (default)

---

## PRINT\_TEMPLATE

**Prerequisite** : DEVICE, JOB\_TYPE (=PRINT)

**When to use** : When you need to create dynamic content for your disc labels, or use the same print files with multiple printer types without having to track printer specific PRN files.

**When not to use** : When using a print image that has already been created (i.e. a PRN file) or the job does not include printing

**See Examples** : 6,7,8

See also : PRINT\_BACKGROUND, PRINT\_ENTRY\_#, PRINT\_QUALITY, PRINT\_MEDIA

Description:

R-Quest Print Template files (.rpt) are produced using the Label Designer within TrueNet FX API. Print Templates allow for dynamic text and backgrounds to be created, and also allow for print files to be created that match the unique properties of an installed printer, i.e. printer x/y offsets, print brightness, thermal strobe settings etc. These parameters are set on the Network Device, and checked by TrueNet FX API prior to producing the PRN file used in the job.

Template files remain independent of the actual printer type (unlike PRN files that are printer type specific), and TrueNet FX API will create the appropriate temporary PRN file for the installed printer when the job is created. This allows the use of a single template (.rpt) file to be used across multiple systems, each using a different printer type, and greatly simplifies print file maintenance.

---

## **PUBLISHER**

**Prerequisite :** DEVICE, JOB\_TYPE (=BUILD), BUILD\_TYPE, BUILD\_PATH

**When to use :** When building an ISO or UDF file system disc from scratch.

**When *not* to use :** When using a disc image that has already been created, or print only jobs.

**See Examples :**

### **Description:**

Publisher is a text element of an ISO9660 volume descriptor, and is a place holder for publisher information. This field is optional, but when used has a maximum 128 characters.

---

## **QUANTITY**

**Prerequisite :** DEVICE, JOB\_TYPE

**When to use :** Whenever you need more than one copy

**When *not* to use :** This is optional is the QUANTITY=1

**See Example : 2**

### **Description:**

Indicates how many discs are required for this job. This parameter is optional, and if not present the quantity defaults to 1. Valid quantities are 1-9999 decimal.

---

## **SELECT\_MEDIA**

**Prerequisite :** DEVICE, JOB\_TYPE=PRINT (print only)

**When to use :** When creating a print only job

**When not to use :** When creating a job that includes Build, Copy or Compare

The SELECT\_MEDIA setting is only used when running a print-only job. For all other job types the media type is inferred from the disc image and this setting will be ignored.

Media types supported (subject to installed recorders and license type) are:

```
SELECT_MEDIA=CD  
SELECT_MEDIA=DVD  
SELECT_MEDIA=DVD_DL  
SELECT_MEDIA=BD  
SELECT_MEDIA=BD_DL
```

Note that the media type must be present in the system for the job to run.

---

### **TRACK\_# (where # is a decimal number)**

**Prerequisite :** DEVICE, JOB\_TYPE (=BUILD), BUILD\_TYPE=AUDIO\_CD

**When to use :** When building an audio CD from WAV files.

**When not to use :** When creating a data disc or when using a disc image that has already been created, or print only jobs.

**See Example :** 11

**IMPORTANT :** Files must be CD-Ready 44.1khz 16 bit stereo WAV files.

#### **Description:**

TRACK\_# is used to specify the source file for each audio CD track, where # is the track number. Tracks must be numbered consecutively incrementing by 1. If a track number is missing, no more tracks are added. e.g.

```
TRACK_1=c:\MyFirstTrack.wav  
TRACK_2=c:\MySecondTrack.wav
```

Up to 99 tracks can be added to an audio CD subject to available space on the disc.

---

### **VOLUME**

**Prerequisite :** DEVICE, JOB\_TYPE (=BUILD), BUILD\_TYPE, BUILD\_PATH

**When to use :** When building an ISO or UDF file system disc from scratch.

**When not to use :** When using a disc image that has already been created, or print only jobs.

**See Examples :**

#### **Description:**

Volume is a text identifier for an ISO9660 or UDF volume descriptor. This is the Volume name normally displayed by an Operating System when a disc is inserted.

---

## **WRITE\_SPEED**

**Prerequisite :** DEVICE, JOB\_TYPE (=BUILD orCOPY)

**When to use :** When writing discs

**When *not* to use :** When not writing discs

**Parameters :** A decimal number (without the 'x') for write speed

**See Example : 11**

Description:

The Write speeds it the maximum write speed to request when writing this image. Note that this is not a guaranteed number, and the recorder may use a slower speed depending on the media inserted. For example, setting a WRITE\_SPEED=16 (for 16x) for DVD would request a 16x write speed, but if only 4x media had been inserted then the recorder would default to 4x instead.

---

# Additional Status Added To Command Files By TrueNet FX API

Once TrueNet FX API has read and accepted the command file, additional information will be added to the file, and where appropriate updated during the course of a job. For example, if the job calls for more than 1 disc to be produced, the current “Accepted” and “Rejected” count will be updated to the file while the job is running, so that a user application can read this information for ‘near real time’ job status. You should not modify any of these

The added identifiers are:

## **ACCEPT**

This indicates the number of discs in this job completed so far.

## **REJECT**

This indicates the number of discs in this job that were rejected.

## **STATUS**

This indicates the current status of this job. This is updated approximately every 5 seconds, so provides near-real-time status. See Appendix D for status codes.

## **JOB\_HANDLE**

This is used for internal processing only, and should not be changed. It is also required for automatic recovery of API jobs in the event of power failure or other instances where TrueNet FX API application was not able to maintain communication with the system.

## Remote Device Status Files (.rds)

TrueNet FX API will maintain a Remote Status File for each known Device on the network. You should not attempt to write to this file, and should consider it read-only. You should also poll this file only as often as is absolutely required, because having this file open (even for read-only) may prevent TrueNet FX API from updating it. During an update the file may appear to be empty, and retries may be required. If you delete this file it will be recreated by the TrueNet FX API application.

A typical file could look like this :

```

DEVICE_NAME=R-QUEST NS2100
DEVICE_IP=192.168.7.148
FIRMWARE=6.04.05
ONLINE=YES
JOBS_QUEUED=1
JOBS_RUNNING=1
JOBS_DONE=0
HOPPER_COUNT=4
HOPPER_0_MEDIA=DVD
HOPPER_0_TYPE=Input
HOPPER_0_COUNT=0
HOPPER_1_MEDIA=DVD
HOPPER_1_TYPE=Input
HOPPER_1_COUNT=32
HOPPER_2_MEDIA=Any
HOPPER_2_TYPE=Output
HOPPER_2_COUNT=1
HOPPER_3_MEDIA=Any
HOPPER_3_TYPE=Output
HOPPER_3_COUNT=0
RECORDERS=2
CD_RECORDERS=2
DVD_RECORDERS=2
BD_RECORDERS=0
BD_DL_RECORDERS=0
RECORDER_0_ACTIVITY=Writing
RECORDER_0_PROGRESS=55
RECORDER_1_ACTIVITY=Idle
RECORDER_1_PROGRESS=0
PRINTER=FlashJet 2
PRINTER_CONSUMABLES=RQ56=54% RQ57=95%
SYSTEM_STATUS=OK
HDD_SPACE=74598

```

An explanation of each of the entries follows.

## **DEVICE\_NAME**

The name given to the device by the administrator. This can be changed by the administrator by logging on to the system using the Web Browser, accessing the Administration dialog, Network tab and setting a new name. All systems on the network MUST have a unique name in order to use the API. The API references systems by their names.

The DEVICE name is retrieved from the system, and matched to any API command files created in the API directory.

## **DEVICE\_IP**

The IP address that is being used to access this system

## **FIRMWARE**

This is the current firmware version running on the network device. All devices MUST be running the same firmware version. Running systems with mixed firmware versions is **not** supported.

## **ONLINE**

Used to indicate if the system is currently online (“ONLINE=YES”) and able to process jobs or offline (“ONLINE=NO”) and unable to process jobs. The Online status of a system is controlled via the keypad.

## **JOBS\_QUEUED**

## **JOBS\_RUNNING**

## **JOBS\_DONE**

The number of jobs currently queued within the system, currently running in the system and currently in a ‘done’ (completed) state. Note that the number of jobs queued may not always be the number of jobs queued via the API. As an example, if other users have queued jobs via the Web Browser GUI, those jobs will be reported within these numbers. In addition, the system may schedule it’s own internal maintenance tasks, which also show up in these numbers.

Where jobs are only ever scheduled via API command files the JOBS\_DONE count should always return to 0 within a few seconds, because ‘Done’ jobs are automatically deleted from the system by the API (and the command files renamed to ‘.don’).

## **HOPPER\_COUNT**

The number of hoppers within a system depends on the system being used. R-Quest NS4500s will show 6 hoppers, whereas NS2100s will show 4 hoppers and 7000 & 9000 series systems show only 1 hopper (because TrueNet sees the entire capacity as a single hopper).

**HOPPER\_#\_MEDIA****HOPPER\_#\_TYPE****HOPPER\_#\_COUNT**

When operating an NS4500 or NS2100 system the hoppers may be assigned to different media types and to different functions such as Input, Output etc. Each hopper will have 3 entries that indicate the current media type (CD, DVD, DVD DL, BD, BD DL), the current I/O mode (Input, Output, Reject) and the number of discs calculated to be present. Note that this calculation is approximate and is based on the standard thickness of media. Where media thickness varies (i.e. a thicker Inkjet coating) the count is not guaranteed to be accurate, but will generally be within one or two discs when the hopper is full. Accuracy will increase when the hopper has fewer discs.

You should not attempt to schedule a job for a media type unless the system contains input hoppers with that media.

Media types and hopper functions can be changed by the administrator by logging on to the system using the Web Browser and choosing the Hoppers tab in the Administration dialog.

**RECORDERS****CD\_RECORDERS****DVD\_RECORDERS****BD\_RECORDERS****BD\_DL\_RECORDERS**

The total number of recorders within a system is noted in the RECORDERS line, together with the number of recorders (included in the total) capable of each media type. Note that currently not all Blu-ray recorders are dual-layer capable, so the BD\_DL\_RECORDERS may not always be the same as BD\_RECORDERS.

You should not attempt to schedule a job for a media type unless the system includes recorders capable of writing that media type.

**RECORDER\_#\_ACTIVITY**

Each recorder is tracked for its activity type within the system. The possible activity types are: "Writing", "Reading", "Lead-In", "Lead-Out", "Comparing", "Auditing", "Disabled".

**RECORDER\_#\_PROGRESS**

The progress of a recorder during "Writing", "Reading" and "Comparing" is tracked and updated to the status file approximately every 5 seconds. When a recorder is not in one of these states the progress indicator is invalid. The progress is a decimal number indicating the percent complete.

**HDD\_SPACE**

The amount of free space on the system's internal hard disk. Do not attempt to send images larger than the available space. The amount of space is shown as a decimal number in megabytes.

**PRINTER**

The model of printer installed on the system is indicated by this entry.

**PRINTER\_CONSUMABLES**

The type and approximate levels remaining of the printer consumables is indicated by this entry.

**SYSTEM\_STATUS**

The current state of the system is updated approximately every 5 seconds. Possible values are :

<b>Error Message</b>	<b>Cause Information</b>
<b>OK</b>	The system state is normal
<b>INPUT HOPPER EMPTY</b>	The system input hopper is empty and requires refilling
<b>OUTPUT HOPPER FULL</b>	The system output hoppers are full - empty them.
<b>MAIL SLOT HOPPER FULL</b>	The system mail slot output is full - empty it
<b>JOB ABORT</b>	The system encountered a fatal job error
<b>PRINT CLAMP FAILED</b>	The printer failed to clamp the media and this may require operator attention
<b>PRINTER ERROR</b>	The system encountered a printer error that it was unable to recover from. This usually required operator attention
<b>DOOR IS OPEN</b>	An operator has opened the door. All jobs will pause until the door has been closed
<b>DOOR WAS OPENED</b>	An operator has previously opened the door, but the door is now closed. The system will begin it's automatic hopper checking process
<b>CHECKING HOPPERS</b>	Following a door open, the system is now checking the hoppers

<b>Error Message</b>	<b>Cause Information</b>
<b><i>PRINTER SUPPLIES OUT</i></b>	The printer consumables supply is exhausted and jobs have been paused. This will require operator attention
<b><i>POSITIONING ERROR</i></b>	The robot encountered a positioning error. It is likely that this will require operator attention
<b><i>ERROR INSERTING PRINTER</i></b>	The system encountered an unrecoverable error while attempting insert a disc in to the printer, and all automated retries have failed. This will require operator attention.
<b><i>ERROR INSERTING DRIVE</i></b>	The system encountered a problem inserting a disc in to a recorder, and automated retries have failed. This will require operator attention
<b><i>ERROR GETTING DISC FROM PRINTER</i></b>	The system encountered a error while picking a disc from the printer, and all automated retries have failed. This will require operator attention
<b><i>ERROR GETTING DISC FROM DRIVE</i></b>	The system encountered a error while picking a disc from a recorder, and all automated retries have failed. This will require operator attention
<b><i>ERROR LOADING FROM INPUT</i></b>	The system encountered a problem picking a disc from the input hopper and all automated retires have failed. This is different from an Input Hopper Empty state, and will require operator attention
<b><i>ACTIVATE CARTRIDGE</i></b>	A new InkJet cartridge is required, and must be activated at the system keypad
<b><i>CHECKING PRINTER</i></b>	The system is currently checking the printer status. This is not an error state
<b><i>ERROR #</i></b>	An error internal error (#) has occurred and there is no text descriptor available. Check the system LCD or the system log file (using the Web Browser interface) for more information

# Appendix A

## Building / Merging ISO / UDF File Systems

Building an ISO file system is usually very straight forward, requiring only the BUILD job type and BUILD\_TYPE=ISO\_CD (or ISO\_DVD) coupled with the BUILD\_PATH.

When creating a number of unique discs based upon the same core file system, it is often the case that most of the files remain the same, with only a single file, or a small number of files that change. This requires the entire image to be built and *sent* to the duplicator before any changes can be made to the file system for the next build. This can be both time consuming and difficult to program.

MERGE files can resolve this problem by allowing the 'static' core file system to remain in one place, while creating the unique files elsewhere, and merging them as the target ISO file system is built. This allows many jobs to be scheduled at the same time, and each unique ISO file created independently of the others. Take the following example local hard disk file system; the core file system is held within the **MyIsoDir**, with the unique files held in separate **MyTempNumbers** and **MyTempPics** directories. The root of the target ISO image is within the **MyIsoDir** directory, so the **MyTempNumbers** and **MyTempPics** directories will not be included.

MyIsoDir

Directory

SerialNum

MyTempNumbers

SN\_10000

Serial.txt

SN\_10001

Serial.txt

MyTempPics

PIC\_1000.JPG

PIC\_1001.JPG

If the files in the SerialNum directory are to be unique on the target disc, build the above file system using the **MyIsoDir** as the BUILD\_PATH. Now MERGE the files from another directory (e.g. **MyTempNumbers\SN\_10000**), and target them to the **\Directory\SerialNum** so that they appear on the final image in that position. Use the Build and Merge commands as follows:

```
JOB_TYPE=BUILD+COPY
BUILD_TYPE=ISO_CD
BUILD_PATH=c:\MyIsoDir
MERGE_SOURCE_1=c:\MyTempNumbers\SN_10000
MERGE_TARGET_1=\Directory\SerialNum
```

Note that **MyIsoDir** was not specified in the MERGE\_TARGET command, because the root of the ISO file system that is built comes from within the **MyIsoDir**.

Using the above example, the final ISO file system would look like this:

```
Directory
    SerialNum
Serial.txt
```

Placing unique files in their own temporary directories allows the building of identical named files but with unique content, which can be merged with the main files when the ISO file system is built, without affecting the other unique files.

It is also possible to use multiple merge commands, simply increment the number following the MERGE\_SOURCE and MERGE\_TARGET commands by one each time.

Example:

```
JOB_TYPE=BUILD+COPY
BUILD_TYPE=ISO_CD
BUILD_PATH=c:\MyIsoDir
MERGE_SOURCE_1=c:\MyTempNumbers\SN_100000
```

```
MERGE_TARGET_1=\Directory\SerialNum  
MERGE_SOURCE_2=c:\MyTempPics\PIC_1001.JPG  
MERGE_TARGET_2=\Directory
```

Note that the directory specified by the MERGE\_TARGET command must already exist. The MERGE\_SOURCE can be either a single file, or a directory. When a directory is specified, all files and sub directories within that merged directory will also be merged.

Use MERGE\_TARGET\_#=\ to merge files to the root of the target ISO file system, where # is the merge number.

# Appendix B

## Example Command Files

Following are some example command files to show how to use the API.

### Example 1:

Make a single copy of an ISO CD image with a ready built PRN (print) file. Note that many optional parameters are not included, e.g. QUANTITY, because the default value (i.e. 1) is what was required.

```
DEVICE=MyDevice
JOB_TYPE=COPY+PRINT
IMAGE_FILE=c:\Images\image.iso
IMAGE_TYPE=RAW_CD
PRINT_FILE=c:\PrintFiles\image.prn
```

### Example 2:

Simulate 5 copies of an RQI image.

```
DEVICE=MyDevice
JOB_TYPE=COPY
IMAGE_FILE=c:\Images\image.rqi
IMAGE_TYPE=RQI
QUANTITY=5
BURN=NO
```

### Example 3:

Build and copy an ISO 9660 CD, with a custom print label (3 replacement fields), compare the disc after writing. Also shows the use of the comments fields.

```
#Set Device Type
DEVICE=MyDevice
#Set Build Type
JOB_TYPE=BUILD+COPY+COMPARE+PRINT
BUILD_TYPE=ISO_CD
BUILD_PATH=c:\MyBuildDirectory
```

```
# Set the Print information
PRINT_TEMPLATE=c:\Prints\MyLabel.rpt
PRINT_ENTRY_1=My Line 1
PRINT_ENTRY_2=Line 2 Here
PRINT_ENTRY_3=This is Line 3
```

#### **Example 4:**

Print a single disc using a pre-built PRN print file.

```
DEVICE=MyDevice
JOB_TYPE=PRINT
PRINT_FILE=c:\PrintFiles\PrintImage.prn
```

#### **Example 5:**

Print 1 disc from a Template file – no fields replaced

```
DEVICE=MyDevice
JOB_TYPE=PRINT
PRINT_TEMPLATE=c:\PrintFiles\PrintFile.RPT
```

#### **Example 6:**

Build and copy an ISO 9660 DVD with a merged file (e.g. Serial number etc) in to a directory called \Install\SN, produce a custom print label (3 replacement fields), and compare the disc after writing. Also shows the use of the comments fields.

```
#Set Device Type
DEVICE=MyDevice
#Set Build Type
JOB_TYPE=BUILD+COPY+COMPARE+PRINT
BUILD_TYPE=ISO_CD
BUILD_PATH=c:\MyBuildDirectory
# Merge the following file
MERGE_SOURCE_1=c:\SnFiles\SN356465.txt
MERGE_TARGET_1=\Install\SN
# Set the Print information
PRINT_TEMPLATE=c:\Prints\MyLabel.rpt
PRINT_ENTRY_1=My Line 1
PRINT_ENTRY_2=Line 2 Here
PRINT_ENTRY_3=This is Line 3
```

**Example 7:**

Print 1 Disc from a Template file with 2 dynamic fields and the background replaced.

```
DEVICE=STATION_1
JOB_TYPE=PRINT
PRINT_TEMPLATE=c:\PrintFiles\PrintFile.rpt
PRINT_ENTRY_1="Field 1"
PRINT_ENTRY_2="Field 2"
PRINT_BACKGROUND=c:\images\background.jpg
```

**Example 8:**

Copy from a DVD ISO with Compare after Write and Print from a Template with 3 fields replaced.

```
DEVICE=MyDevice
JOB_TYPE=COPY+COMPARE+PRINT
IMAGE_FILE=C:\Images\SampleDVD.iso
IMAGE_TYPE=RAW_DVD
PRINT_TEMPLATE=c:\PrintFiles\PrintFile.RPT
PRINT_ENTRY_1="This is Field 1"
PRINT_ENTRY_2="This is Field 2"
PRINT_ENTRY_3="This is Field 3"
```

**Example 9:**

Build and copy an ISO9660 CD disc from the c:\source directory, and fixate the copies. Request the system to 'fast start' the recording as the image arrives.

```
DEVICE=MyDevice
JOB_TYPE=BUILD+COPY
BUILD_TYPE=ISO_CD
BUILD_PATH=c:\source
FIXATE=YES
FAST_START=YES
```

**Example 10:**

Build and copy an ISO9660 DVD disc from the c:\source directory, and do not fixate the copies.

```
DEVICE=MyDevice
JOB_TYPE=BUILD+COPY
BUILD_TYPE=ISO_DVD
BUILD_PATH=c:\source
```

FIXATE=NO

**Example 11:**

Build an audio CD with 4 tracks and sets the write speed to 16x.

```
DEVICE=MyDevice
JOB_TYPE=BUILD+COPY
BUILD_TYPE=AUDIO_CD
TRACK_1=c:\Audio\Track1.wav
TRACK_2=c:\Audio\SecondTrack.wav
TRACK_3=c:\Audio\TrackThree.wav
TRACK_4=c:\Audio\04-Track.wav
WRITE_SPEED=16
```

# Appendix C

## API Command File Error Codes

If the command file is rejected, an error code will be added to the command file - together with a plain text description of the error type. The file will also be renamed with a '.bad' extension.

**ERROR\_CODE** This is a numeric value as detailed in the following table. Example: ERROR\_CODE=1

**ERROR\_TEXT** This will be an ASCII description of the error.  
Example:  
ERROR\_TEXT=DEVICE NOT FOUND

A full list of the error codes is:

<b>CODE</b>	<b>ERROR_TEXT</b>	<b>Likely Cause</b>
0	NONE	If the file has been renamed to .err and there is no error specified, check the 'STATUS' line for more information. Also, check the system status file (.rds) for more information. It is likely that there is a system problem (e.g. input hopper empty) that is causing a delay to this job, rather than an error specific to this job. See SYSTEM_STATUS
1	DEVICE NOT FOUND	The device specified in the DEVICE= statement was not found on the system. Check the name against those on the system.
2	OPTION NOT SUPPORTED	There are some reserved keywords (for possible future feature expansion) that TrueNet FX API will report as unsupported.

<b>CODE</b>	<b>ERROR_TEXT</b>	<b>Likely Cause</b>
3	BAD IMAGE FILE TYPE	An unknown or incompatible image file type was specified.
4	BAD FIXATE PARAMETER	Only YES or NO are valid.
5	IMAGE FILE NOT FOUND	The disc image file could not be found.
6	PRINT FILE NOT FOUND	The print image file could not be found.
7	TEMPLATE FILE NOT FOUND	The print template could not be found.
8	UNSUPPORTED BUILD TYPE	TrueNet FX API is unable to produce dynamic print files for the installed printer type.
9	PRINTER IS NOT DYNAMIC	TrueNet FX API does not have an integrated printer driver for the installed printer type. You need to use pre-built PRN files with this printer.
10	INVALID PRINT QUALITY	The parameters for the print quality are invalid. See PRINT_QUALITY command for more information.
11	INVALID BUILD PATH	The supplied BUILD_PATH is not valid.
12	BACKGROUND FILE NOT FOUND	Unable to find the specified PRINT_BACKGROUND file. Check the path and file name.
13	INVALID BACKGROUND FILE	Unable to use the supplied file type as a background file. Try using an RGB (not CMYK) JPEG, GIF, TIF, PNG etc.
14	NO_AUDIO_TRACKS	No audio tracks were found, but the build type was set to AUDIO_CD

<b>CODE</b>	<b>ERROR_TEXT</b>	<b>Likely Cause</b>
15	BAD AUDIO	The audio track is not the required type. Audio files must be CD ready 44.1Khz 16bit Stereo WAV files.
16	REMOTE_DISC_IMAGE_NOT_FOUND	Unable to locate the requested remote disc image
17	REMOTE_PRINT_IMAGE_NOT_FOUND	Unable to locate the requested remote print image.
18	DEVICE_NOT_NETWORK_MODE	The device is not in network mode. Put the device in network mode.
19	RECOVERY_FAILED_NO_JOB	An attempt was made to recover a job when restarting TrueNet FX API. However, there is no job on the remote system that matches this job.
20	IMAGE_FILE_TOO_SHORT	The image file is too short and will fail to write correctly. For CDs, the shortest ISO file size is 600KB (4 seconds of data).

# Appendix D

## API Job Status Codes

STATUS	Explanation
Building Print	TrueNet Fx API is currently building the PRN file from the supplied print Template File (.rpt)
Check Media	Check the media type in the system for compatibility with this job type.
Check Printer	Check the printer status in the system. There may be a fault.
Done	The job is done
Error	The job encountered an error for which there is no pre-defined error message. Check the system logs.
Error, bad disc image	The system tried to start this job but the disc image may be corrupt. This is a fatal error for this job, and the job may need to be deleted from the system manually (using the browser interface).
Error, bad print image	The system tried to print from the supplied image, but detected that the print file was bad. This is a fatal error for this job, and the job may need to be deleted from the system manually (using the browser interface).
Error: No Valid Media	There is no media configured in the system for this job type, e.g. this job requires DVDs but there are no DVD input hoppers configured in the system.
Getting Disc	TrueNet FX API is currently retrieving a disc image from the system.
Getting Print	TrueNet FX API is current retrieving a print image from the system.
Input Empty	The input hoppers are empty
Killing	The job was 'killed' by a user (via the Web Browser interface) and the system is in the process of aborting the job. See 'Stopping' for more information.

<b>STATUS</b>	<b>Explanation</b>
Lost	The job was lost, probably due to network connectivity issues.
Low Media	The system currently has low media in the input hoppers. Refilling the hoppers is recommended.
No Media Input	There is currently no input hopper configured for this job
No Media Output	There is currently no output hopper configured for this job
NO RECORDERS	There are no recorders enabled on the system. If the recorders have been set to auto disable after a number of consecutive rejected discs, eventually the system may disable all the recorders and report this condition.
Output Full	The output hoppers are full. Remove discs from the output hoppers to allow the job to continue.
Paused	The job was paused
Pausing	This job is pausing (will result in the job being 'Paused')
Queued	The job is currently on the network device and queued for production. As soon as it is the next job to be done the status should change to 'Running'.
Reading	Currently Reading a master disc
Resuming	A job that had previously been stopped is now resuming.
Running	The job is currently running.
Scheduled	The job has been parsed by the TrueNet FX API parser and is currently waiting to be sent to the system. This will typically be the case when the system's job queue is already full.
Sending Disc	TrueNet FX API is currently sending the disc image to the system.
Sending Print	TrueNet FX API is currently sending the print image to the system.
Stopped	The job has been stopped (by a user). Use the Web Browser interface to restart the job.

<b>STATUS</b>	<b>Explanation</b>
Stopping	A user has told the system (via the Web Browser interface) to stop the job. The system is in the process of stopping the job, but the job is not yet stopped. This would be the case if the discs had not yet been removed from the recorders or printer. The difference between 'Stopping' and 'Killing' a job is that 'Stopping' a job will allow any discs currently being recorded (or printed) to continue until completed, but no more discs will be loaded. "Killing" a job aborts the job immediately and places all discs currently in the recorders (as part of 'this' job) in to the reject hopper.
Unknown	The job encountered an error for which there is no pre-defined error message. Check the system logs.

# Appendix E

## Running from the Command Line

It is possible to launch the FX API program via the command line on most systems. This is particularly useful for system that support Java, but for which there is no ready made installer.

To launch the program from the command line, ensure that you are in the directory that contains the `truenet_fx_api.jar` file then run the following command:

```
java -jar -Xmx512m -Xms500m truenet_fx_api.jar
```

This will launch the TrueNet FX API program with a default GUI that allows the user to select the IP address (or URL) of the system to connect to.

## Headless Operation

If your server is running headless (no monitor) or is not running a GUI (e.g. X11) then you should add the following command line options :

Command	Description
<code>/Q</code>	Running in Quiet mode and does not show a GUI
<code>/IPxxx.xxx.xxx.xxx</code>	<code>xxx.xxx.xxx.xxx</code> is the IP address of the system to connect to. e.g.  <code>/IP192.168.7.136</code> Substitute 192.168.7.136 in the above example with the IP address of your system.

When using the command line switches, these should be added after the jar file, e.g.

```
java -jar -Xmx512m -Xms500m truenet_fx_api.jar /Q /IP192.168.7.136
```

If the program is launched in Quiet mode (`/Q`) and no `/IP` is specified the program will exit automatically, returning an error code of -1.

## Manually configuring the API devices

Running the FX API via the command line in headless mode means there is no way to configure the remote devices using a GUI configuration tool. If you have the opportunity to run the GUI version for configuration purposes this is the recommended method. However, if your server does not have a graphical display (e.g. an X11 server on Linux) then the remote systems must be configured manually.

The configuration file is called TrueNetFX\_Settings.fxs and is usually found in the user's home directory. You should run the FX API program at least once so that it creates this file for you. Once created you can edit the file to include the following settings:

```
# Number of devices to use - there must be an IP for each device
# In this example there are two devices. If you have only one device the
# you should have 1 as the parameter
DEVICE_COUNT=2

# An IP address for each device is required. The first device is numbered 0 (zero)
# and increment the number by 1 for each device. Do not skip numbers.
# Your IP addresses are likely to be different from this shown in this example
DEVICE_IP_0=192.168.7.136
DEVICE_IP_1=192.168.7.141

# A path to the API directory is required. This is the directory that the API job
# files will be placed. Without this information FX API has no way to pick up jobs
API_PATH=C:\api
API_TMP_PATH=C:\api

# This parameter controls whether FX API can transfer images to multiple devices
# at the same time. While this may reduce the effective throughput to an individual
# device, it may allow an overall improvement than running sequentially. To transfer
# images sequentially (i.e. only transfer to one system at a time, set this to NO.
PARALLEL_TRANSFER=YES

# If you want the FX API to log it's operations add these lines to the configuration file
# replacing the "c:\Logging" with the directory to save the logs to. A new log file will
# be created each day.
LOGGING_DIR_ENABLED=1
LOGGING_DIRECTORY=C:\Logging
```

Once the configuration file has been saved, restart (or run) the FX API program. The FX API program will only read these configurations once at start up, and any changes made are not used until the program is restarted.

