

# FileXfer

## File Transfer Jobs

Raymond E. Marcil  
<marcilr@gmail.com>

Revision 0.0.1 (June 28, 2016)  
379340b

### Abstract

The FileXfer application is a system for automated file transfer jobs for copying files. “There are 3 applications that make up the usage collection framework: `filexfer`, which does the actual file transfers; `filexfer-jobmonitor`, which is configured to monitor various aspects of jobs and create NMS alarms when necessary; and `filexfer-dataloader`, which bulk-loads file data into database tables. There are also housekeeping scripts called `filexfer-filearchive`, which keeps files in the data directory pruned and compressed, and `filexfer-fileunarchive`, which allows files to be pulled out of the archive so filexfer jobs can work with them again.”<sup>1</sup>

---

<sup>1</sup>Usage Collection Framework (filexfer)

# Contents

<b>Contents</b>	<b>2</b>
<b>List of Figures</b>	<b>3</b>
<b>List of Tables</b>	<b>3</b>
<b>List of Definitions and Abbreviations</b>	<b>4</b>
<b>Introduction</b>	<b>5</b>
<b>Job Scheduling</b>	<b>5</b>
<b>Application Logging</b>	<b>5</b>
<b>File Transfer Logging</b>	<b>5</b>
<b>Dataloader</b>	<b>6</b>
<b>Examples</b>	<b>7</b>
Escaping < and > Symbols . . . . .	7
Enumerate . . . . .	7
Comments . . . . .	7
Footnotes . . . . .	8
Hyperlinks . . . . .	8
Table Examples . . . . .	8
Verbatim . . . . .	10
Version Number . . . . .	11
<b>Endnotes</b>	<b>12</b>
<b>Appendix</b>	<b>12</b>

List of Figures

1	File and Directory Structure . . . . .	10
---	--	----

List of Tables

1	EASEMENTS_17B Table . . . . .	8
2	USS XML index elements . . . . .	9
3	Demo . . . . .	9
4	Daemons . . . . .	9
5	XREF_MTR_QMQ Table . . . . .	9

## List of Definitions and Abbreviations

- **MOA** - Municipality of Anchorage

## Introduction

The FileXfer system...

## Job Scheduling

Jobs are scheduled using a web interface at `nms.operations.gci.com/relevance`. Navigate to the “FileXfer” application and click the “File Transfer Jobs” link. Job execution happens on `prod-prov4-cdr1.operations.gci.com`. A cron job executes every minute from `/etc/cron.d/filexfer` to kick off the various `filexfer` scripts.<sup>2</sup>

## Application Logging

The `filexfer` applications log to the `/var/log/filexfer` directory on `prod-prov4-cdr1.operations.gci.com`. The parent `filexfer` jobs log to `filexfer-get.log` and `filexfer-put.log`. The `jobmonitor` and `dataloader` applications log to `jobmonitor.log` and `dataloader.log`. The `filexfer` applications log to the `/var/log/filexfer` directory on `prod-prov4-cdr1.operations.gci.com`. The parent `filexfer` jobs log to `filexfer-get.log` and `filexfer-put.log`. The `jobmonitor` and `dataloader` applications log to `jobmonitor.log` and `dataloader.log`, respectively. Each file transfer job is executed as a child process and gets its own log file. The format is `filexfer-{neName}-{idJob}-{get,put}.log`.

By default, the jobs log at the warn level. Adjust the level to info to get a high-level view of the application’s state. Adjust log verbosity by modifying the appropriate config file in `/etc/filexfer`. The changes will take effect after the next program execution.

Errors are also logged to a database table which can be browsed in the `filexfer` web interface under the ‘Logs & Errors’ view. This view includes messages logged at `warn`, `error`, and `fatal` severity.<sup>3</sup>

## File Transfer Logging

Every file transfer is recorded in a database table. There are two reasons for this table: first, it tells `filexfer` which files have already been transferred, and second, it provides an audit trail for SOX compliance. The table is `filexfer.logs` on `sadc-cdr-mysql1.operations.gci.com`. Use the `filexfer.joblogs` view to easily find logs by job name or network element ID.

File transfer logs may also be viewed in the ‘Logs & Errors’ page of the web interface.<sup>4</sup>

---

<sup>2</sup>Usage Collection Framework (`filexfer`)

<sup>3</sup>Usage Collection Framework (`filexfer`)

<sup>4</sup>Usage Collection Framework (`filexfer`)

## Dataloader

Dataloader jobs are configured using the web interface at `nms.operations.gci.com/relevance`. Navigate to the “FileXfer” application and click the “Data Load Jobs” link. These jobs are executed every minute as long as there are files in the load queue.<sup>5</sup>

---

<sup>5</sup>Usage Collection Framework (filexfer)

## Examples

Series of useful L<sup>A</sup>T<sub>E</sub>X markup. Need to break out to separate examples.tex file.

## Escaping < and > Symbols

To get \$<\$ or \$>\$ just wrap the symbols in \$ for math mode.

## Enumerate

1. DNR - Alaska State Department of Natural Resources
  - HI - Historical Index, not maintained since 1982
  - LE - Land Estate, maintained by SGU
  - ME - Mineral Estate, maintaind by SGU
2. Alaska State Surveys
  - ASBLT - As-Built Survey
  - ASCS - Cadastral Survey

## Comments

**COMMENTS** Comment — *Sean Weems, Spring 2003*

We should get the **COMMENTS** column searchable via the landrecords application before we do much anything else – shouldn't be too hard.

*Errata: Plats spanning multiple sections*

A few anomalies can be observed in the **AKPLATS** table. Specifically plats exist that span multiple sections. Since the table only has a single column, **SCODE**, that accepts a single section code, SGU (Status Graphics Unit) has handled this problem by entering multiple rows in the table, each with a different section that point to the same plat or file. Multiple section plats are indicated by setting the **TCODE** column to the value 37, and making an appropriate notation like *Section 24-25-26-27* in the **REMARKS** column.

[FIXME: Perhaps the **SCODE** column should accept an array of sections?]

## Footnotes

See my footnote<sup>6</sup> generated with:

```
\footnote{\href{http://www.google.com/search?q=latex+footnotes}
{Search google for footnotes.}}
```

GoogleGuide — Linking to Search Results.<sup>7</sup>

## Hyperlinks

Use `\href{}` to generate hyperlinks:

```
\href{http://www.google.com}{Google}}
```

Yields: [Google](http://www.google.com)

## Table Examples

Column Name	Type	Description
EQS	VARCHAR2(1)	!NULL map shows village selections
ITM_COL	VARCHAR2(1)	USGS ITM column: 1-6
ITM_ROW	VARCHAR2(1)	USGS ITM row: A-E
QMQ_ABBR_DNR	VARCHAR2(3)	Three character DNR abbreviation for the QMQ
RASTER_FILENAME	VARCHAR2(50)	Physical path to file
RASTER_PATHNAME	VARCHAR2(50)	URL path to PDF of map
SCODE	VARCHAR2(2)	Supplement map code: 1,2,3,...
COMMENTS	VARCHAR2(256)	Plat comments

Table 1: EASEMENTS\_17B Table

---

<sup>6</sup>[Search](#) google for footnotes.

<sup>7</sup>GoogleGuide — [Linking to Search Results](#).



XML element	Description
FNUM	US Survey file number
MERIDIAN	BLM meridian code 12 = Copper River 13 = Fairbanks 28 = Seward 44 = Kateel 45 = Umiat
TOWNSHIP	Five character Township code
RANGE	Five character Range code
PAGE	Survey page number 1,2,3,...
FILENAME	Relative path to file in direcorey

Table 2: USS XML index elements

col 1	col 2	col 3	col 4
item 1	item 2	item 3	item 4
item 1	item 2	item 3	item 4

Table 3: Demo

Virtual Machine	Apache	ELM	LM	Elluminate Server
dcs-elive-prod01		x	x	x
uaa-elive-dev01	x	x	x	
uaa-elive-server01				x
uaa-elive-prod01		x	x	x
uaf-elive-prod01		x	x	x
uas-elive-prod01		x	x	x

Table 4: Daemons

Column Name	Type	Description
MTR	VARCHAR2(9)	Meridian, Township, Range, example: <i>C026S054E</i>
QMQ	VARCHAR2(3)	Quarter Million Quadrangle code, example: <i>DIL</i> (Dillingham quadrangle)

Table 5: XREF\_MTR.QMQ Table

## Verbatim

“The verbatim environment is a paragraph-making environment that gets L<sup>A</sup>T<sub>E</sub>X to print exactly what you type in. It turns L<sup>A</sup>T<sub>E</sub>X into a typewriter with carriage returns and blanks having the same effect that they would on a typewriter.”<sup>8</sup>

```
\begin{verbatim}
  text
\end{verbatim}
```

### Figure formatting with verbatim

The following figure leverages verbatim for proper formatting:

```
gis/raster/
  dnr/
    map_library/
      plats/
        SP/YYYYMMDD/*.pdf          # indexed
        HI/YYYYMMDD/*.pdf          # Indexed
        ASLS/YYYYMMDD/*.pdf        # Indexed
      recorded-plats/
        YYYYMMDD/*.pdf
  blm/
    easements_17b/YYYYMMDD/*.pdf   # indexed
    mtp/YYYYMMDD/*.pdf             # non-indexed
    usrs/YYYYMMDD/*.pdf             # indexed
    usrs-notes/YYYYMMDD/*.pdf       # indexed
    uss/YYYYMMDD/*.pdf             # indexed
    uss-notes/YYYYMMDD/*.pdf        # indexed
    usms/YYYYMMDD/*.pdf             # indexed
    usms-notes/YYYYMMDD/*.pdf       # indexed
  usgs/
    drg/
      collared/
        250K/
        63K/
        25K/
        24/
      decollared/
      tools/
      missing\_data/
    dem/
    doq/
    topo/
```

Figure 1: File and Directory Structure

## Version Number

It is often desirable to add a version number to a document for tracking or revision control. CVS or Subversion users can use the `rcs` or `svnInfo` packages for inline version information.

[FIXME: Need to complete details here]

## Endnotes

1. L<sup>A</sup>T<sub>E</sub>X verbatim  
<http://www.kfunigraz.ac.at/binder/texhelp/ltx-79.html>

## Appendix

A Guide to L<sup>A</sup>T<sub>E</sub>X  
<http://www.astro.rug.nl/kuijken/latex.html>

L<sup>A</sup>T<sub>E</sub>X - From Wikibooks, the open-content textbooks collection  
<http://en.wikibooks.org/wiki/LaTeX>

L<sup>A</sup>T<sub>E</sub>X Notes  
[http://luke.breuer.com/time/item/LaTeX\\_Notes/180.aspx](http://luke.breuer.com/time/item/LaTeX_Notes/180.aspx)