

# Ad Loading On Penfold

## Baseview Feeds (via E-Mail Push)

**Step 1:** Baseview feeds are sent via e-mail to push.morris.com

E-mail must include exactly one attachment and the email format must be:

To: [adloader@push.morris.com](mailto:adloader@push.morris.com)  
Subject: ADLOAD:*filename*  
Or  
Subject: ADLOADER:*filename*  
Body: Property name\*

Filename must match the name of the attachment.

Property name must match the property directory name used on alexis for loading.

(example: for /var/spool/adloader/*examiner.net*/classifieds/incoming the property name is examiner.net)

Note: The /home/adloader/ad\_feeds/*propertyname*/data and home/adloader/ad\_feeds/*propertyname*/logs directories must be created manually before the first e-mail is sent to push.

**Step 2:** Push.morris.com has a proc-script run by adloader which strips off the attachment from the e-mail, validates that it is in a correct format and dumps it into a staging area folder on push. (/home/adloader/ad\_feeds/*propertyname*/data). The file is renamed *filename.txt.#####*, where ##### is a unique time stamp.

This process generates the following log files:

/home/adloader/logs/loader_mail_log.txt	Output of Perl script
/home/adloader/logs/errors.txt	Extraneous prints from Perl script
/home/adloader/logs/errors2.txt	Error message from Perl script

For e-mails for quarterhorsesnews.com barrelhorsesnews.com, an additional script is run to convert the text file to the correct format.

**Step 3:** The user adloader has a cron which fires every 15 minutes to run this script:

/home/adloader/scripts/get\_new\_ad\_feeds.sh

This script pull files, via FTP, from the staging directories on push and dumps them into:

/var/spool/adloader/*propertyname*/classifieds/incoming.

Uses the /home/adloader/scripts/get\_new\_ad\_feeds.cfg file to determine what type of transfer to do. Currently set up as: ASCII Mode, Get All Files, Remove Files After get.

NOTE: No addition files need to be modified for the scraper to pull files from push. However, you must have the /var/spool/adloader directory set up for the property.

**Step 4:** Root runs another script, /usr/local/bin/process\_bv\_ads.sh, at 12:30am, 1:30am, 2:30 am, 6:30 am, 8:30 pm and 10:30 pm (also 7:30 am for Holland) which loads the data into the database and moves the incoming files into:

```
/var/spool/adloader/propertyname/classifieds/data
```

This script calls /usr/local/bin/load\_bv\_ads.sh for each property.

```
/usr/local/bin/load_bv_ads.sh "Property Name"
```

## **perplexes**

### **Mail-Push Error Messages:**

#### **Error MP-101: Missing Attachment**

Your ad load attempt failed.  
Reason: Missing attachment named '\$destFileName'.

#### **Error MP-102: Missing Named Attachment**

Your ad load attempt failed.  
Reason: Missing attachment named '\$destFileName'.

#### **Error MP-103: Illegal File Name**

Your ad load attempt failed.  
Reason: Illegal file name (\$destFileName). '\$1' not allowed.

#### **Error MP-104: Too Many Attachments**

Your ad load attempt failed.  
Reason: You sent too many attachments. Only one allowed.

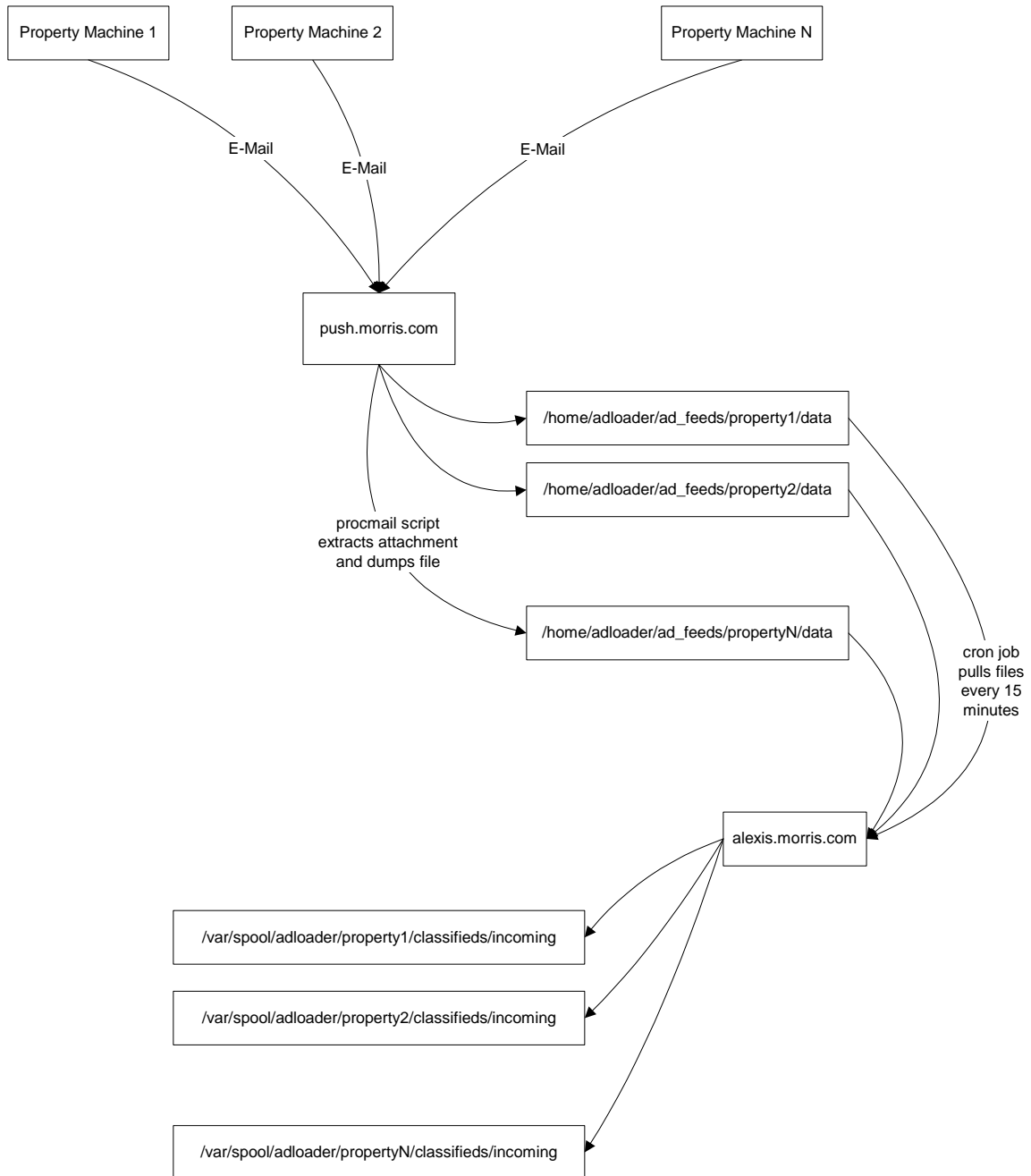
#### **Error MP-105: Date Not Specified**

Your ad load attempt failed.  
Reason: START DATE and/or END DATE not specified.  
Error:(\$!)  
Call support.

#### **Error MP-106: Attachment Copy Failed**

Your ad load attempt failed.  
Reason: Could not copy attachment to \$destFilePath.  
Error:(\$!)  
Call support.

## Ad E-mail Push Process



## Baseview Feeds (via FTP Pull)

**Step 1:** Properties FTP their ad feeds to their staging area FTP sites.

**Step 2:** The user display on alexis fires off a cron job every 15 minutes which runs this script:

```
/home/adloader/scripts/get_the_ad_feeds.sh
```

This script reads the `/home/adloader/scripts/get_the_ad_feeds.prop` file to determine which properties to pull. It includes the FTP information for each FTP site.

The script also uses `/home/adloader/scripts/get_the_ad_feeds.cfg` to determine the transfer mode, currently set up as: ASCII Mode, Get All Files, Remove Files After get.

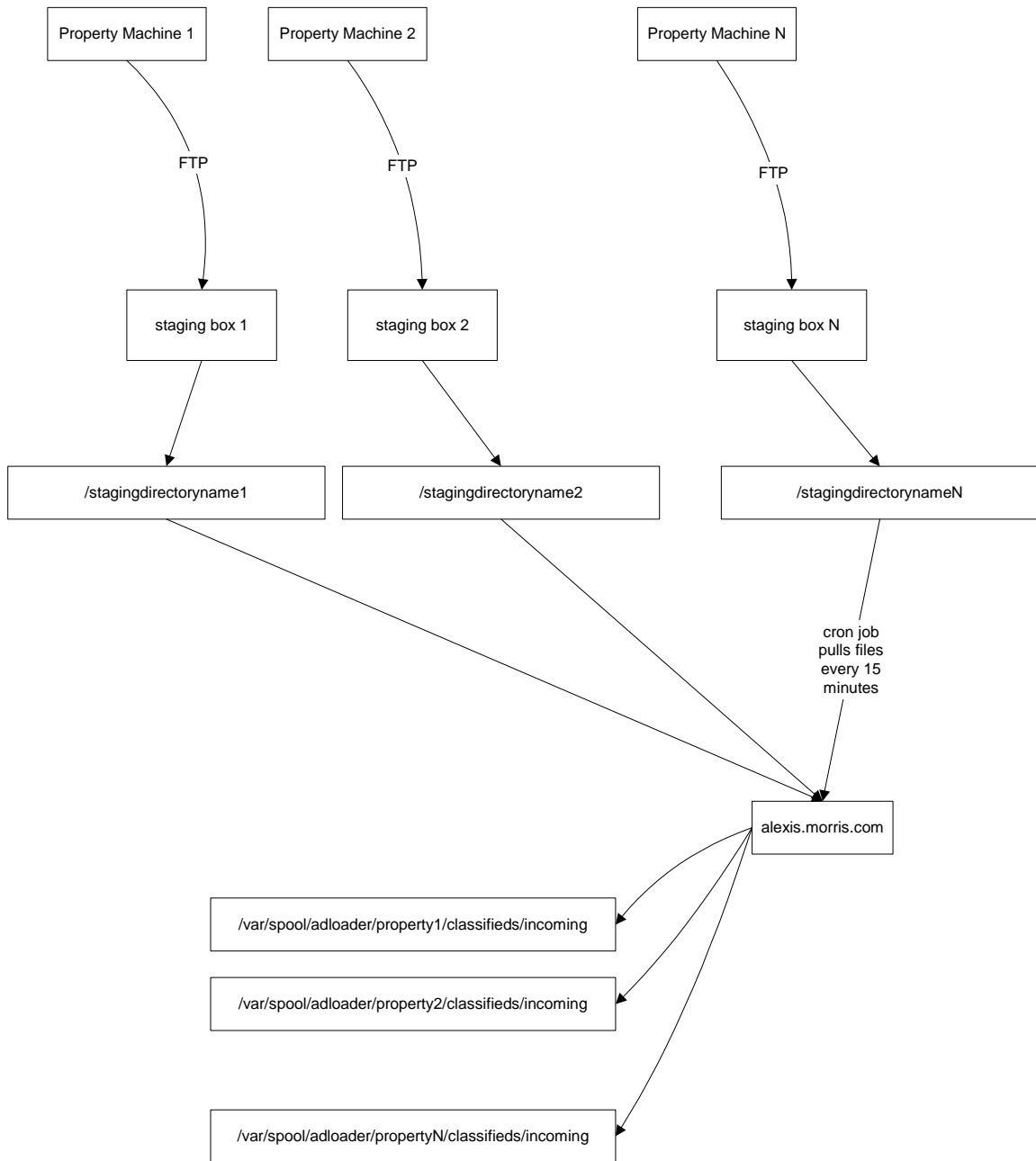
**Step 3:** Another script, `/usr/local/bin/process_bv_ads.sh`, runs at 12:30am, 1:30am, 2:30 am, 6:30 am, 8:30 pm and 10:30 pm (also 7:30 am for Holland) which loads the data into the database and moves the incoming files into:

```
/var/spool/adloader/propertyname/classifieds/data
```

This script calls `/usr/local/bin/load_bv_ads.sh` for each property.

```
/usr/local/bin/load_bv_ads.sh "Property Name"
```

## Ad FTP Pull Process



## **Crontab on Alexis**

Run as root

```
30 0,1,2,6,20,22 * * * /usr/local/bin/process_bv_ads.sh
```

#Added by TLGilman to support Holland's request to have ads processed later  
4/8/2004.

```
30 7 * * * /usr/local/bin/process_bv_ads.hol.sh
```

Run as adloader

```
5,20,35,50 * * * * /home/adloader/scripts/get_new_ad_feeds.sh
```

Run as display

```
10,25,40,55 * * * * /home/adloader/scripts/get_the_ad_feeds.sh
```

```
0,15,30,45 * * * * /opt/pdfdisplay/bin/display.pl
```

**Not sure yet what /opt/pdfdisplay/bin/display.pl is other than it is used to load old version of display ads**

## **Relevant Information**

push.morris.com

username: adloader

password:load!me

The loader runs each day. Baseview currently does not send ad numbers, so if you run the loader twice, the ads will be inserted twice.

## DTI Feeds

**Step 1:** DTI servers have a cron job that runs nightly to upload their data feeds. The feeds themselves are named #####.xml where ##### is CCYYMMDD. While the file is has an extension of .xml it is not an XML file.

The cron job logs into alexis, uploads the xml file, exits, logs in again and uploads a series of .pict files.

The files are uploaded to the `/var/spool/property_name/classifieds/dti` directory.

Some properties upload a file called #####.raw in addition to the .xml file. This file is appended onto the end of the .xml file and then run through the parser.

Those properties that run the latest version of DTI software send .eps files (hi-res images) rather than the .pict files. Currently this list is believed to be limited to Athens and Augusta.

**Step 2:** A cron runs once per night for each of the properties. All of the properties do not run at the same time. They are broken up by using different scripts:

```
process_dti_ads.sh
process_dti_ads2.sh
process_dti_ads3.sh
process_dti_ads_nondaily.sh
```

Note: For Jacksonville, I have created yet another version of the parser to accommodate their need for fielded data. The new mdParser.jar file located under `/opt/MDW/bin/new/`

**Step 3:** The images (either .eps or .pict) are converted to a .jpg format and are transferred to the web server for the property by the shell script `/usr/local/bin/convert_dti_logos.sh` which is called directly by the `/usr/local/bin/load_dti_ads.sh` script.

The .eps and .pict files are converted using the script `/usr/local/bin/logo.sh` called from `convert_dti_logos.sh`

The images are moved to the appropriate web server based on the property attribute `LOGOS_ABS_DIR` via rcp.

### Logging:

```
/tmp/load_dti_ads.$$
This file is e-mailed to the users.
```

## Cron jobs for running DTI ads

```
30 19 * * * /usr/local/bin/process_dti_ads.sh
#Ad loading for DTI properties w/ obits (except Jacksonville)
30 21 * * * /usr/local/bin/process_dti_ads2.sh
#Ad loading for Lubbock (since they can't meet their obits deadline)
0 23 * * * /usr/local/bin/process_dti_ads3.sh
#Ad loading for non-daily DTI publications
30 22 * * * /usr/local/bin/process_dti_ads_nondaily.sh
```