HETDEX Data Release 2:
Emission Line eXplorer (ELiXer)
User's Guide

Introduction

The Emission Line eXplorer (ELiXer) is a diagnostic/debugging tool that, along with a limited API, is included as part of the HETDEX data release. ELiXer does not perform detections on its own, but primarily aggregates HETDEX observation data and external photometric catalogs to facilitate the examination of emission line detections and aid in line classification.

Caveats

Runtime Environment

ELiXer was originally developed and tested against Python 2.7 but has been fully converted to Python 3.7.x.

ELiXer is a single threaded at the application layer, however there are multi-threaded packages included (such as emcee).

ELiXer does support bulk execution on several Texas Advanced Computing Center (TACC) supercomputing clusters (stampede2 and wrangler) via SLURM batch scheduling.

The base memory footprint is relatively small (~ 200 MB), however, some photometric imaging files are very large and, even with lazy loading, the memory requirements can briefly exceed 15 GB and, depending on the timing of garbage collection, execution with the Python kernel may (rarely) encounter memory allocation issues. In key areas, ELiXer attempts to detect this condition and implements random sleeps and limited retries in simple attempt at remediation.

Line Classification

Classification, where possible, is supplied by the ELiXer tool and consists primarily as two independent approaches. First, ELiXer, in turn, assigns the detected emission line each of two dozen potential classifications (see table below) and examines the spectra for the other lines from the same set (fitting (by least squares) a Gaussian to the position where the other line(s) would be expected (allowing for some error and velocity offsets). All combinations are scored by SNR, line width, integrated line flux, and offset from the expected line center. At this time, absorption features are not used. If exactly one solution scores more than 50% of the total score of all solutions and is greater than a minimum acceptable score, it is marked as the probable classification via this method. In all other cases, no classification is assumed.

Second, a Bayesian posterior ratio of the probability that the line is Lyα to the probability that the line is [OII], [noted generally as P(LAE)/P(OII)] is computed based on Leung et al 2016 (Leung, Andrew S, et al 2017, "BAYESIAN REDSHIFT CLASSIFICATION OF EMISSION-LINE GALAXIES WITH PHOTOMETRIC EQUIVALENT WIDTHS", arXiv:1510.07043v2). This does not examine the possibility of any other line classifications and only compares P(Lyα) to P([OII]) on the line flux and estimated equivalent width. The continuum estimate for the equivalent width comes from several sources and, as such, a P(LAE)/P(OII) ration is produced for each source of the continuum estimate. The first continuum estimate comes from the HETDEX extracted 1D spectra (near the emission line), but with a magnitude limit ~24, this is often an upper limit on the continuum. A second estimate is made using the entire HETDEX spectrum passed through an SDSS g-band filter (with a magnitude limit of ~ 24.5). When photometric imaging is available, additional continuum estimates are made using preferentially a g-band filter (or r-band, if g-band is not available). One estimate comes from a forced circular aperture placed at the HETDEX position. That aperture radius starts near the average seeing for the catalog and grows in 0.1" steps until the growth flattens (or the growth matches the sky noise) (note: each step is recorded in the ELiXer HDF5 catalog file, see HDR2 Data Model and Application Programming Interface document, section 9). Another estimate comes from the best matched Source Extractor elliptical aperture (where the selection is made for the aperture that overlaps the HETDEX position whose center is nearest the HETDEX center or, if there are none, the aperture whose edge is nearest the HETDEX position, with 1.5"). Lastly, continuum estimates are made for up to the top three (nearest, within, by default, 3") photometric catalog reported detections using the catalog's reported magnitude (again, in g-band preferentially) for the detection. No validation is performed on the catalogs and the reported magnitude is taken at face value. The separation to the HETDEX detection does not include an estimate of the spatial extent of the detections and again assumes point-sources. Each of these P(LAE)/P(OII) ratios is computed and reported independently. Where no catalog detections or imaging is available or the code does not converge on a ratio, none is provided.

After all continuum estimates are made, a single, combined estimate is produced from a weighted and inverse variance average of all contributing estimates. That combined continuum value is then used to produce a combined, single PLAE/POII value that is displayed in the top left of the report.

The combined PLAE/POII value is then included as one factor, along with the presence of additional emission lines and the physical size of the detection (from the imaging and assuming the various possible redshifts with a concordance cosmology) to produce a single P(LAE) value [0.0-1.0] as an approximate probability that this detection is an LAE. This is very experimental, and, although it shows good results in testing, should not be fully trusted and the user is strongly recommended to review each detection report in its entirety before accepting a classification.

Lyα (1216)	Hβ (4863)	NV (1241)	NaI (4980,5153)
OII (3727)	Hγ (4342)	SiII (1260)	CaII (3935)
OIII (4960,5007)	Hδ (4102)	HeII (1640)	
CII (2326)	Hε (3970)	NeIII (3869,3967)	
CIII (1909)	Hζ (3889)	NeV (3347)	
CIV (1549)	Hη (3835)	NeVI (3427)	

Table 1 ELiXer Emission Lines (to nearest Å)

The photometric imaging catalogs currently used by ELiXer come from a variety of sources and instruments and have not been adapted or calibrated for HETDEX needs. Please see the README under the imaging directory for a description of the imaging catalogs possible limitations (`/work/03946/hetdex/hdr2/imaging/README`).

Installation

ELiXer supports a local pip installation, which should also install the necessary Python packages, but it is assumed that the user has already installed Python3 and hetdex_api (see HDR2 Data Model and Application Programming Interface document and brief instructions below). In addition to the main application, ELiXer also exposes a limited API (described later) for accessing the imaging catalogs and PLAE/POII calculations.

Recommended Setup

If you have not installed `hetdex_api`, clone that repository from github:
https://github.com/HETDEX/hetdex_api.git

`cd` into the cloned directory and execute this command:

```
> pip install --user --upgrade -e .
```

This will pip install `hetdex_api` (overwriting any previous `hetdex_api` installation) and setup an auto-install update whenever a new `git pull` is executed on `hetdex_api`.

In a separate directory, clone ELiXer from github:
https://github.com/HETDEX/elixer.git

It is recommended that you select the 'hdr2' branch for this release or the 'master' branch for the latest, stable release.

`cd` into the cloned directory and execute this command (to select 'hdr2'; if 'master' is wanted, do not execute the command (or replace `hdr2` with `master`):

```
> git checkout hdr2
```

Then execute this command:

```
> pip install --user --upgrade -e .
```

This will pip install `elixer` (overwriting any previous `elixer` installation) and setup an auto-install update whenever a new `git pull` is executed on `elixer`.

It is recommend that you periodically execute `git pull` on both `hetdex_api` and `elixer`, to stay up to date if you are using the 'master' branch.

The pip installation process should install the other required packages, though there are sometimes errors. It is simplest to test the installation by invoking the 'help' on ELiXer and then install or update any packages for which ELiXer reports an error.

To do so, execute:

```
> python <path to elixer>/elixer.py --help
```

notice: in the <path to elixer>, elixer.py is under another subfolder, also named elixer, so if you cloned the repository to ~/code/elixer the command would look like this:
> python ~/code/elixer/elixer/elixer.py --help

The packages that may need a manual installation include:

- Common Python packages (should already be included in Python3 w/o additional installation): numpy, matplotlib, pylab, argparse, sys, os, distutils, glob, shutils, socket
- Less common packages: (install with "pip install --user xxx" where xxx is the package name): astropy, configparser, pandas, photutils, scipy, and tables
- One additional HETDEX specific package, pyhetdex, is also required: ("pip install --user --extra-index-url https://gate.mpe.mpg.de/pypi/simple/ pyhetdex")

Running ELiXer (generating reports) on TACC

TACC Environment Caveats

At the time of this writing, TACC environments still default to Python2. It is recommended that you place the following lines in your .bashrc file (usually after "SECTION 3" if that comment exists):

```
module unload python2
module unload xalt
module load intel/18.0.2
module load python3
```

Even with these changes, you must still invoke the Python3 Interpreter explicitly with: `python3`

Running ELiXer

Because HDR2 already maintains ELiXer reports for all its detections (see `hetdex_api` notebook `15-ELiXer_Report_DB_Access` for examples), there should generally be no need to run ELiXer again to generate a report. However, should you need to do so, ELiXer is intended to be run in one of two modes ... single instance and batch (via SLURM). If you are running on your own desktop, you must use the single instance mode, but if you are running on a TACC cluster you should normally use the batch or SLURM mode (TACC discourages the execution of high cost tasks from the login nodes).

The basic command line for the two modes are almost identical, with the SLURM mode supporting a few additional parameters related to the SLURM mechanism.

In the single instance mode, ELiXer will serially process each provided detection. In the SLURM mode, ELiXer will spawn multiple instances of itself, dividing the detections across the instances, but still processing serially within any given instance.

Run time varies with server, load, and proximity to the data, but you can generally assume approximately one minute per detection report.

You may launch the ELiXer process with a Python call: e.g.

(single instance version)

```
> python3 <path to elixer>/elixer/elixer.py --<options>
```

-or-

(SLURM version: notice '`selixer.py`' instead of '`elixer.py`')

```
> python3 <path to elixer>/elixer/selixer.py --<options>
```

The "--`help`" switch will print to screen a simple break out of the command line options. Also NOTE that the SLURM version will NOT spawn multiple instances if the "--`help`" switch is on the command line (in this case, `elixer` and `selixer` are equivalent (the other cases are with the `merge`, `merge_unique`, `upgrade_hdf5`, or `prep_recover` switches described later).

Common Usage Examples

Although there are many options, ELiXer is anticipated to be used in only a few ways with HETDEX Data Release 1. Essentially, you will either provide an RA, Dec and search radius or a list of detection IDs and ELiXer will produce a report for each.

The following examples will assume the SLURM version invocation. For compactness, `selixer` represents the follwing:

```
python3 /<path to elixer>/elixer/selixer.py
```

EXAMPLE 1

```
> selixer --recover --ra 150.025406 --dec 2.087600 --error 3.0 --name
example1 --tasks 0 --email yourname@utexas.edu
```

Here an --ra and --dec are provide (in decimal degress ... however, hms and dms notations will also work, e.g.: --ra 10h00m6.10s --dec 2d05m15.36s are equivalent).

--recover is a switch that instructs ELiXer to run each detection to completion before starting the next one. This allows ELiXer to be run a second time, using the exact same command, if the first command timed out in the queue and it will resume where it left off and only process detections for which a report does not yet exist.

--error is the radius in which to search from the given RA and Dec and is ALWAYS in arcsecs.

--name is the output directory name under which the results will be written.

--tasks 0 specifies that ELiXer should set the number of instances to spawn based on the cluster on which it is running. You may override this and supply a non-zero value to force ELiXer to use no more than that number of tasks.

--email is entirely optional, but will generate emails to the supplied address when the SLURM job actually begins (when it exits the wait queue) and when it completes or ends via error.

Additional, optional commonly used command line switches:

--time : supply a maximum hh:mm:ss runtime for the SLURM job (if not supplied, ELiXer will calculate a value)

--queue : specify which queue to use to execute the SLRUM job on the cluster (if not supplied, ELiXer will choose a queue)

Example 2

```
> selixer --recover --tasks 0 --email yourname@utexas.edu --dets
detlist --error 2.5 --name example2
```

Here, --dets detlist refers to a file named detlist that contains a list of detectionIDs, one per line.

Obviously, you need to know the detectionIDs in advance (which may well be the case if you are already working with them). This may also be a comma separated list (no spaces) like:

--dets 2000000318,2000000330,2000150414

Output

If you run the single instance of ELiXer, all the output will be immediately under a directory named for the --name switch.

If you run the SLURM version of ELiXer, there will be a series of dispatch_xxxx directories (where xxx is a number) under the directory named for the --name switch and under each of those will be a log file and a directory named for the `--name` on the invocation call. Under this second nested directory will be the output files for the detections.

The output files (excluding files created due to other options that may be supplied on the command line) at a minimum, will consist of a report PDF (named after the detection) and an HDF5 catalog file. Depending on the supplied options (`--jpeg`, `--png`, `--mini`, `--neighborhood <arcsec>`) there will be other files (.png or .jpg) as well.

The PDF reports graphically represent the basic information about the detection including information on the fiber 2D spectra, the fit of the emission line, the full summed/weighted spectra, photometric imaging (if available) and potential catalog matches (if available).

The HDF5 catalog file (see HDR2 Data Model and Application Programming Interface document, section 9) contains the non-visual information in the report as well as additional details.

The HDF5 catalogs are fragmented under each of the dispatch_xxxx directories, but you can combine them all into one catalog (each) by running `selixer --merge` (or `elixer --merge`) at the parent directory of the dispatch_xxxx folders. Warning! While merging individual catalogs is fast, there is a substantial, per file overhead and merging more than a few dozen files should NOT be done on a login node. As an example, it can take several hours to merge 10,000 small files (where it takes only seconds to merge two multi-GB files).

A description of how to interpret the individual PDF reports is presented later in this document.

Running ELiXer (generating reports) on Your Local Box

You may also wish to run ELiXer directly from your own desktop. This can be useful when running only a few detections and for debugging with ELiXer and/or its API.

The individual runtime may be longer since the data will be remote and accessed via the Internet.

For this option to be effective, you will most likely want to mount the TACC `/work` directory (and often, `wrangler:/data` or `stampede2:/scratch` as a remote file systems using `sshfs`.

Briefly, on a linux OS, to mount a remote file you would:

1) Create a local directory as a mount point. In order to mimic the TACC structure, it is recommend you (on your local machine) use `/work` or make a soft-link as `/work` and point to the mount point of your choice. For HDR2, `/data` is commonly used and is faster than `/work`.
2) Issue the following command (using `wrangler:/data` as an example):
   > `sshfs yourname@wrangler.tacc.utexas.edu:/data ~/tacc/data -C -o ServerAliveInterval=30,ServerAliveCountMax=5,noatime`

   Where:
   `yourname@corral.tacc.utexas.edu:/work` = your login to TACC and the remote directory you want to mount. You may substitute wrangler, stampede2, etc for 'corral'.

   `~/data/work` = the path to your local mount point. In this case, it is under the user's home directory and a soft-link at /work should be created. (e.g. from the root directory (/) issue this command:

   ```
   sudo ln -s ~/data/work /work
   ```

   The other options instruct sshfs to allow compression (-C) and send keep alives every 30 seconds and allow 5 timeout responses before giving up.

To close your mount, issue the following command:

```
fusermount -u ~/tacc/data
```

With the remote file system mounted as above, you can issue ELiXer invocations or use the API with the same paths as if you were running on a TACC cluster.

Using the API

This ELiXer version exposes two limited Python APIs that supports accessing the early photometric catalogs as well as the P(LAE)/P(OII) computation.

If you copied the ELiXer source directory or are referencing the HDR2 source path, you must add that path to your system path so that Python can locate the APIs. To do so, add the following code to the top of your Python script:

```
import sys
sys.path.append(<path to elixer)
import catalogs
import classify
```

If you installed ELiXer (as described earlier in this document), you need only include the following lines in your Python script:

```
from elixer import catalogs
from elixer import classify
```

For basic help on each API, you may execute (from within a Python interpreter):

```
help(catalogs)

help(classify)
```

The `catalogs` package provides access to the photometric imaging data and catalogs as well as basic aperture photometry. The `classify` package provides a wrapped interface to the Bayesian P(LAE)/P(OII) ratio computation.

Example iPython Notebooks are provided under the `notebooks` folder in the `hetdex_api` source directory.

`08-ELiXer_Imaging_Catalogs.ipynb`

`09-ELiXer_Line_Classification.ipynb`

Interpreting the Report

As described at the beginning of this document, ELiXer is a diagnostic/debugging tool and is currently intended only to aid in informing expert line identification. Upcoming enhancements in future data releases will provide more authoritative classifications.

Please see the example PDF reports at the end of this document to match up with the following descriptions:

1. (upper left) Combined P(LAE)/P(POII) and P(LAE) (see Line Classifications section at the top of this document)
2. (upper right) version information – the time stamp when this file was created and the version of ELiXer used to create it
3. (upper left) Basic HETDEX information.
    - HETDEX detect ID number (original file name)
    - Observation ID (DatevShot_detectdID)
    - Primary IFU - identifies the SpecID and IFU Slot ID of the IFU that hosts the highest weighted fiber in the detection
    - RA, Dec - the HETDEX-astrometry weighted centroid position … assumes point source)
    - λ (the observed wavelength center of the detection), FWHM (estimated full width half max of the detection)
    - LineFlux – estimated, integrated line flux (erg/s/cm^2)
    - Cont(n) - estimated continuum (narrow) near the emission line(erg/s/cm^2/Å) … note, HETDEX mag limit ~ 24, so this is often a ceiling
    - Cont(w) - estimated continuum (wide) – using the full spectrum - (erg/s/cm^2/Å) … note, HETDEX mag limit ~ 24, so this is often a ceiling
    - EWr – estimated rest frame equivalent width, assuming Lyα and using the HETDEX estimated line flux and continuum from both the narrow and wide estimates
    - S/N – SNR of the detection, χ^2 – Fit to Gaussian
    - P(LAE)/P(OII) – ratio using the HETDEX EWr and Cont(n) AND the EWr and Cont(w)
    - LyA z – the redshift assuming a Lyα identification , OII z – the redshift assuming an OII identification
    - [optional] Possible line identification IF multiple emission lines are matched
4. (top center-left) Cutouts from the HETDEX dataframe(s) in reverse (dark = high count) that cover the wavelength region and fibers of the emission detection. The top most row is a sum of the lower (up to) four rows. Each of the lower (up to) four rows represents one fiber (multiple exposures may be present), but is approximate to 3 fibers tall on the CCD. The first column is from the reduced science frame (green pixels represent masked cosmic ray strikes). The second column is from the pixel flat and the third is a Gaussian smoothed representation of the first column. The text to the left is a (normalized to 1.0) weight of the fiber (the highest weighted fibers are displayed in descending order) followed by the absolute fiber number (1-448) for the CCD. The text to the right (you may have to zoom in to read it) shows additional data (the distance in arcsec to the detection center, the X,Y position if viewing in DS9, the date_shot_exposure#, and the IFUSlot_Amp_Fiber#). Each plot is centered on the fiber's (interpolated) fractional pixel corresponding to the detected emission line wavelength. The

border is colored to match with panels (4) and (7). Green colored pixels are cosmic ray or other defects.

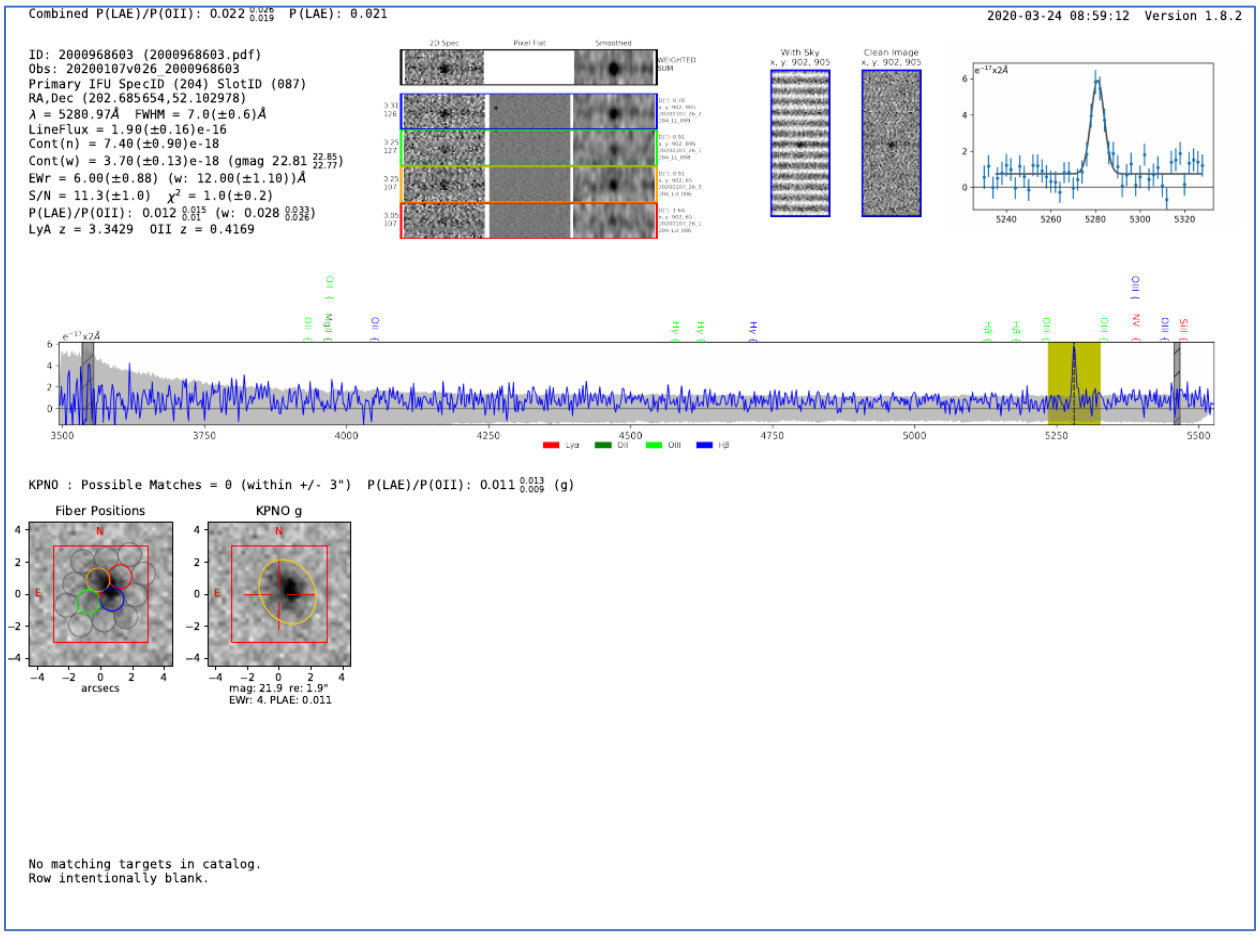
5. (top center-right) A cutout of the CCD region around the main (highest weighted, most central) fiber (the border is blue, color coded to match). This is intended to provide a view on the CCD to check for non-astrometric scattered light or other anomalies. The left of the two images has not had the sky subtracted where the right image has been sky subtracted. The x,y values under the title are the DS9 pixel coordinated of the center of the frame.
6. (top right) A zoomed scatter plot of the weighted and summed 1D spectrum of the detection over plotted with the best fit Gaussian. In the next frame, this is the same region as that highlighted in gold.
7. (center full width spectrum) A full width plot of the weighted and summed 1D spectrum of the detection. The wavelength positions of known strong skylines are highlighted in gray (they should be subtracted from the data and the highlights only indicate a position, not a presence of residual light). If we assume the peak to be the emission line indicated in the legend beneath the plot, then we could see other lines indicated by labels above the plot, color coded to match the legend below. For example, if we assume the emission line to be Lyman Alpha (red) then other lines we might see are indicated by the labels in red. [optional] Colored highlights (other than the gray and gold already mentioned) may be present over additional, possible emission lines. The color matches the associated line classification below the plot (i.e. red = the additional lines are associated with the main line classified as Lyα).
8. (center imaging) Summary of catalog information, including:
    - [optional, if a magnitude was calculated] P(LAE)/P(OII) Bayesian ratio calculated from the HETDEX estimated flux and the continuum estimated from an aperture magnitude (preferentially of g-band) as described in the Line Classifications section at the beginning of this document.
    - The first cutout shows the fiber positions to scale (colors match those in 3, with gray fibers representing other contributing fibers beyond the top four) overlaid on a stacked image (weighted by exposure time). Fiber circles with circumscribed dashed lines indicate fibers that lie on the periphery of the IFU. The red box shows the size of the input error window (the --error parameter) and indicates celestial north.
    - The remaining cutouts represent [optional] supplemental (different catalog) imaging and then one or more available filters (the cutout title identifies the catalog and/or filter). The small colored boxes mark the positions of catalog identified targets that could be the emission line source (based on angular separation to the target center IF it falls within the red search box). Warning! As noted in the Caveats section at the beginning of the document, spatial extent is NOT yet considered. A gold circle or ellipse represents the position and extend of the aperture (see Line Classification section at the top of this document) used to calculate a magnitude (not show is the annulus used to calculate local sky … see the Line Classifications section). Additional SourceExtractor object aperture ellipses are shown in white. If an aperture magnitude is calculated, its value, the (effective) radius of the annulus, the restframe EW (assuming Lyα), and the PLAE/POII ratio is printed below the cutout.
9. (bottom) Summary of information for up to three catalog targets. The colors match the colored boxes in the previous frame (7).

- Separation – angular separation (in arcsecs) between the HETDEX position and the reported (center) position in the catalog. Again, spatial extent is NOT considered at this time.
- 1-p(rand) – a limited estimate of the probability that the catalog target is a random match based on the magnitude of the target and the distribution of similar magnitudes in increasing distance annuli from randomly sampled positions. This is a sorting measure only and is predominantly based on angular distance.
- RA, Dec – the reported RA, and Dec from the catalog
- Spec z – if present, the reported spectroscopic redshift
- Photo z – if present, the reported photometric redshift
- Est LyA rest-Ew – the rest frame equivalent width estimate assuming the line is Lyα with the line flux from HETDEX and the continuum estimate from the catalog
- Est OII rest-Ew – the rest frame equivalent width estimate assuming the line is OII with the line flux from HETDEX and the continuum estimate from the catalog
- mag – the catalog reported magnitude with filter name
- P(LAE)/P(OII) - Bayesian ratio calculated from the HETDEX estimated flux and the continuum as reported from the catalog as described in the Line Classifications section at the beginning of this document.

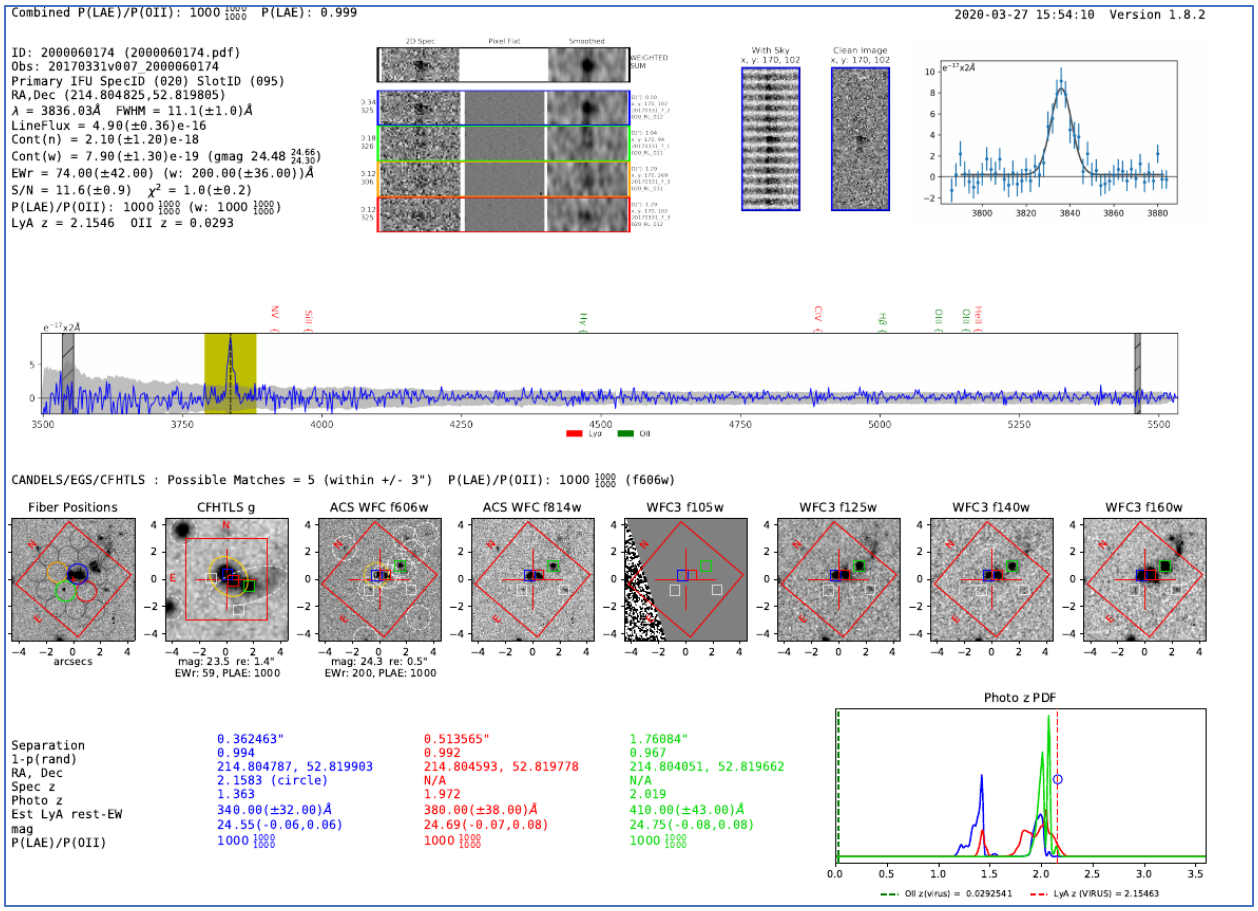
10. [optional] If available a phot-z PDF for up to the top 3 catalog matches. PDF curve colors match the catalog object (blue, red, green). Color matched circles mark spec-z values (if available). Dashed vertical lines represent the redshift if the emission line is OII (green) or Lyα (red).

Some Examples

Probable OII emitter.



Probable LAE (likely an interacting system)



Probable LAE (probably an AGN)