MPDAF- A Python package for the analysis of VLT/MUSE data

L. Piqueras, S. Conseil, M. Shepherd, R. Bacon, F. Leclercq and J. Richard

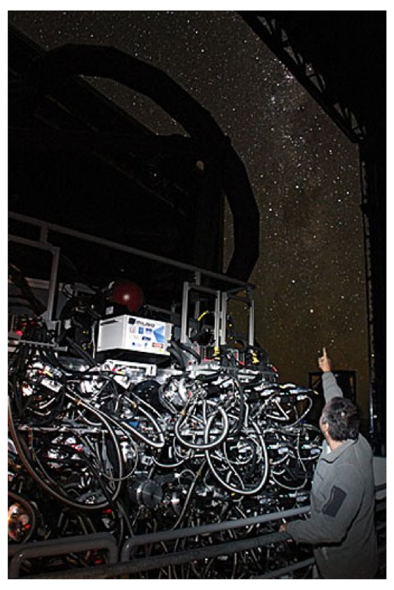
Centre de recherche astrophysique de Lyon(CRAL)

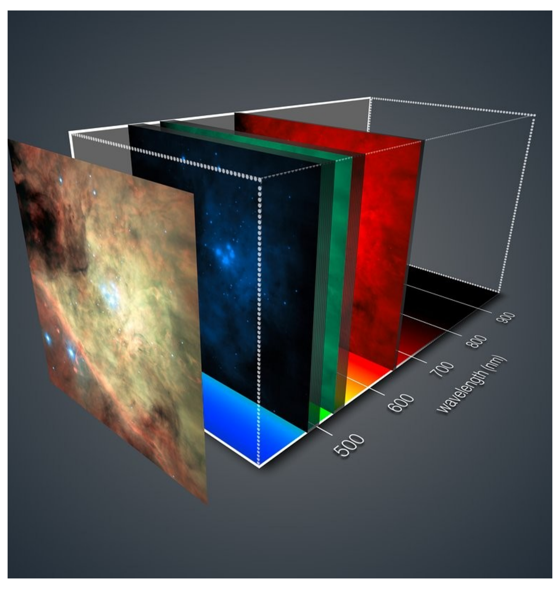
The VLT/MUSE instrument

- Second generation instrument for the VLT
- First light in 2014
- Integral Field Spectrograph (IFS)
- 24 identical spectrographic modules
- Visible range: 465-930 nm
- 90 000 spaxels in a single exposure
- Two main modes:

  WFM: 1'x1' FoV, 0.2"x0.2" spatial sampling

  NFM: 7.5"x7.5" FoV, 0.025"x0.025" spatial sampling





MPDAF: MUSE Python Data Analysis Framework

Installation: `pip install mpdaf`

Requirements: Python 2.7 or 3.3+, Numpy 1.8+, Scipy 0.14+, Matplotlib 0.1+, Astropy 1.0+Nose, Numexpr, CFITSIO, C OpenMP library, Sextractor

Subpackages :

obj Analyze cube, images and spectra
drs Read and manipulate calibration products and intermediate files of the MUSE pipeline
sdetect Detect and manage sources
MUSE MUSE specific tools (slicer numbering scheme, LSF, PSF, mosaic field map

Analysing a Lymanα line in the HDFS MUSE data cube

- Open a MUSE cube and select a sub-cube of 10" around an object

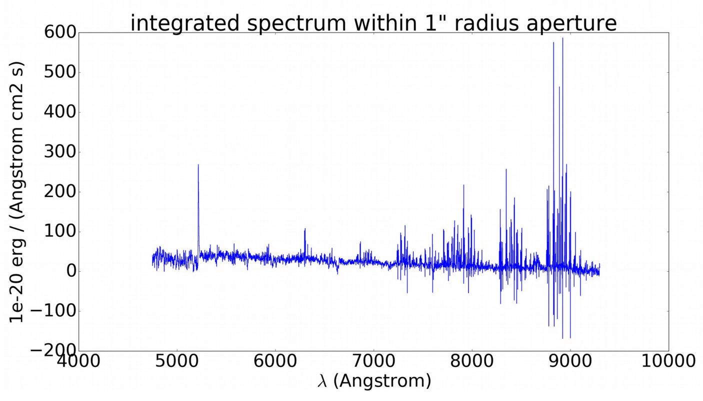
```
In [1]: from mpdaf.obj import Cube
In [2]: cube = Cube('DATACUBE-HDFS-v1.24.fits')
In [3]: cube.info()
[INFO] 3641 x 331 x 326 Cube (DATACUBE-HDFS-1.24.fits)
[INFO] .data(3641 x 331 x 326) (no unit), .var(3641 x 331 x 326)
[INFO] center:(-60:33:48.9321,22:32:55.5267) size:(66.200",65.200") step:(0.200",0.200")
       rot:-0.0 deg frame:ICRS
[INFO] wavelength: min:4750.00 max:9300.00 step:1.25 Angstrom
In [4]: center = (-60.56183, 338.2168)
In [5]: subcube = cube.subcube(center, 10.)
In [6]: subcube.info()
[INFO] 3641 x 50 x 50 Cube
[INFO] .data(3641 x 50 x 50) (no unit), .var(3641 x 50 x 50)
[INFO] center:(-60:33:42.6294,22:32:52.0271) size:(10.000",10.000") step:(0.200",0.200")
       rot:-0.0 deg frame:ICRS
[INFO] wavelength: min:4750.00 max:9300.00 step:1.25 Angstrom
```

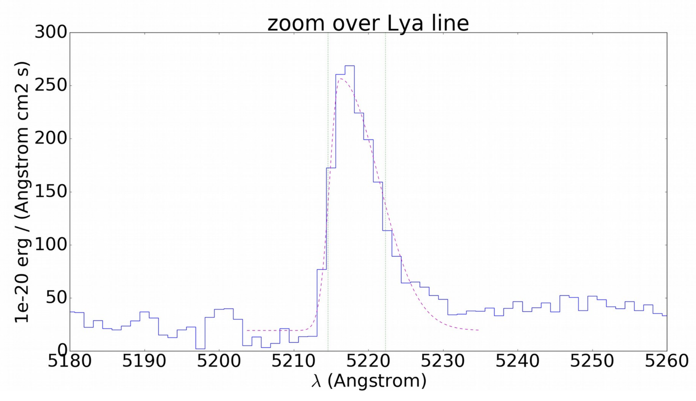
- Compute integrated spectra

```
In [7]: sp = subcube.aperture(center, 1)
[INFO] 100 spaxels sum
In [8]: sp.plot(title='integrated spectrum within 1'' radius aperture')
```

- Zoom over the Lymanα line and perform a fit of the line

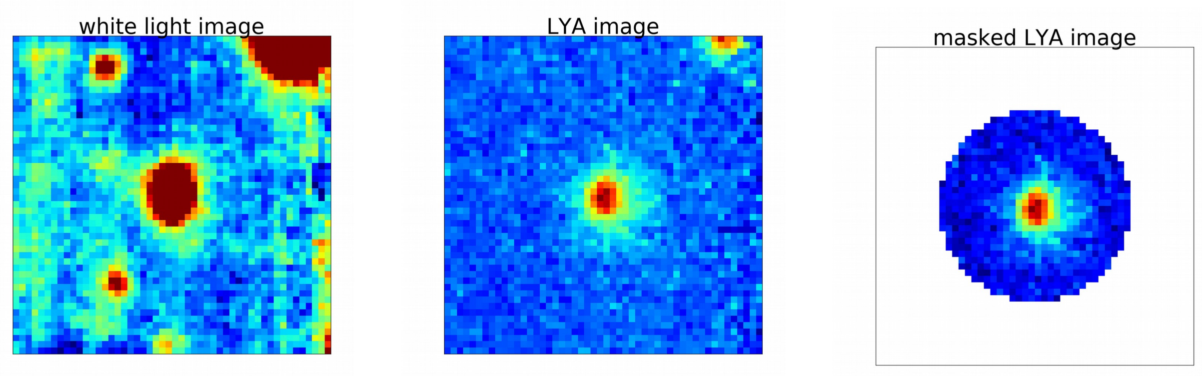
```
In [9]: plt.figure()
In [10]: sp.plot(lmin=5180, lmax=5260, title='zoom over Lya line')
In [11]: lfit, rfit = sp.gauss_asymfit(lmin=5180, lmax=5260, plot=True)
In [12]: lfit.print_param()
[INFO] Gaussian center = 5216.18 (error:0.283944)
[INFO] Gaussian integrated flux = 374.599 (error:86.5779)
[INFO] Gaussian peak value = 119.561 (error:5.69449)
[INFO] Gaussian fwhm = 2.94336 (error:0.574899)
[INFO] Gaussian continuum = 28.3729
In [13]: rfit.print_param()
[INFO] Gaussian center = 5216.18 (error:0.283944)
[INFO] Gaussian integrated flux = 1362.02 (error:86.5779)
[INFO] Gaussian peak value = 119.561 (error:5.69449)
[INFO] Gaussian fwhm = 10.7019 (error:0.667306)
[INFO] Gaussian continuum = 28.3729
In [14]:plt.axvline(lfit.lpeak - lfit.fwhm/2, color='g', linestyle=':')
In [15]:plt.axvline(rfit.lpeak + rfit.fwhm/2, color='g', linestyle=':')
```





- Reconstruct the white-light image and the Lymanα narrow-band image by summing spatial pixels of the cube over the entire wavelength axis or over 11A

```
In [16]: plt.figure()
In [17]: white = subcube.sum(axis=0)
In [18]: white.plot(title='white light image', zscale=True)
In [19]: plt.figure()
In [20]: nb = subcube.get_image(wave=(lfit.lpeak-5.5, lfit.lpeak+5.5))
In [21]: nb.plot(title='LYA image')
```



- Mask some data to easily compute the integrated flux and its error within a 3" radius aperture

```
In [22]: plt.figure()
In [23]: nb.mask_region(center, 3, inside=False)
In [24]: nb.plot(title='masked LYA image')
In [25]: lya_flux = nb.data.sum()*subcube.get_step()[0]*subcube.unit
In [26]: err_lya_flux = np.sqrt((nb.var*subcube.get_step()[0]**2).sum()*subcube.unit**2)
In [27]: print('F_lya(r<3") = %s (error=%0.2f)'%(lya_flux, err_lya_flux.value))
F_lya(r<3") = 445.714263916 1e-20 erg / (Angstrom cm2 s) (error=6.89)
```

Reference: Leclercq and al (in prep)

Offering tools to detect and manage sources

- MUSELET, a Sextractor-based tool to detect emission lines in a data cube

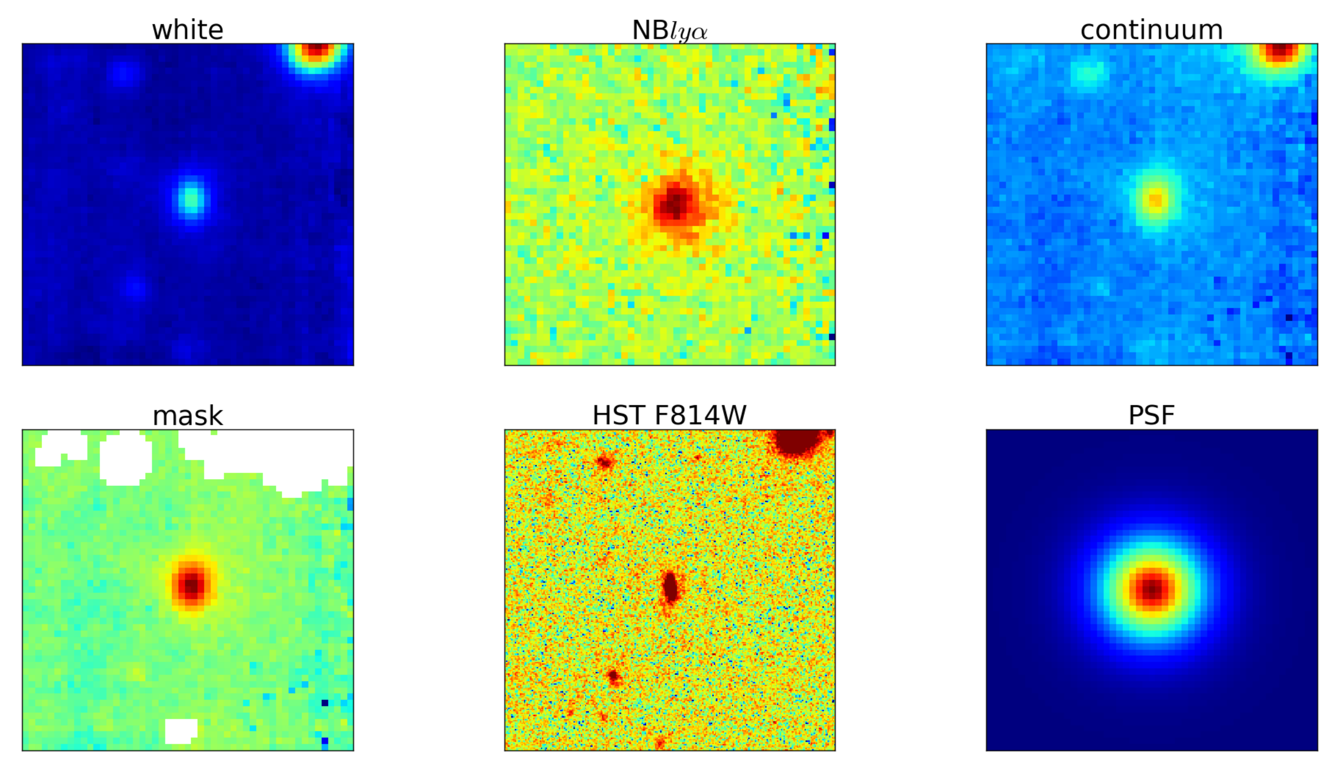
References: Drake et al., 2016, ArXiv e-prints. 1609.02920
Bina & al., 2016, A&A, 590, A14.1603.05833
Bouché & al., 2016, ApJ, 820, 121.1601.07567

- Format to gather all the information on a source in ones FITS file in order to easily exchange information about detected sources. For example, the source FITS file of the Lymanα detected at α=338.2168 and δ=-60.561

```
In [1]: from mpdaf.sdetect import Source
In [2]: src = Source.from_file('source_HDFS_1.24_0043.fits')
In [3]: src.info()
[INFO] ID      =                   43 / object ID
[INFO] RA      =         338.216796875 / RA in degrees
[INFO] DEC     =    -60.56182861328125 / DEC in degrees
[INFO] CUBE    = 'DATACUBE-HDFS-v1.24.fits' / MUSE data cube
[INFO] CUBE_V  = '1.24'               / datacube version
[INFO] FROM    = 'LyAna   '           / detection software
[INFO] FROM_V  = '1.2     '           / version of the detection software
[INFO] SRC_V   = '2.0'                 / source version
[INFO] RA_LYA  =     338.2167821426706 / Ra (deg) of the lya emission centroid by gauss_
[INFO] DEC_LYA =    -60.56187707427184 / Dec (deg) of the lya emission centroid by gauss
[INFO] FWHM    =                  0.74 / full width at half max in arcsec
[INFO] DLYA    =    0.1823430789572579 / Distance (arcsec) of the lya emission centroid
[INFO] 2 spectra: MUSE_EM_LYANA MUSE_TOT_LYANA
[INFO] 11 images: MUSE_CONT_LYANA MUSE_LYA_LYANA MUSE_LYA_MASKED_LYANA HST_F814_LYANA MASK_LYA_OTHERS_LYANA
MUSE_WHITE_LYANA MUSE_PSF_LYANA HST_F814_MASK_LYANA MUSE_CONT_MASKED_LYANA MUSE_CONT_HST_LYANA MASK_OTHERS_LYANA
[INFO] 3 cubes: SUBCUBE_LYANA SUBCUBE_CONT_LYANA SUBCUBE_EM_LYANA
[INFO] 2 tables: SB_TABLE FLUX_TABLE
[INFO] 1 lines
[INFO] 1 magnitudes
[INFO] 1 redshifts
```

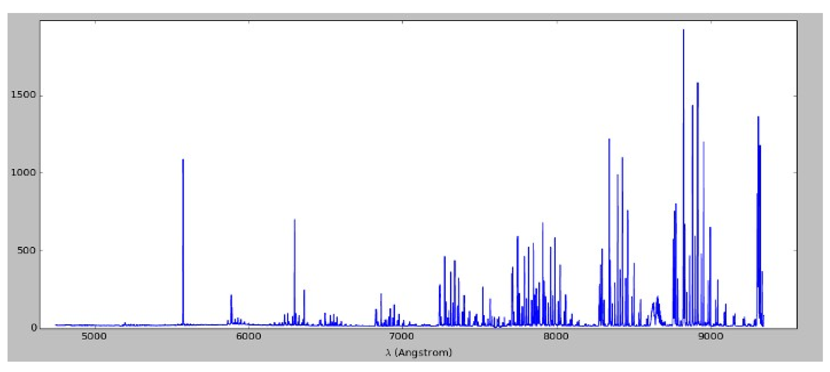
- Lines information, magnitudes and redshift values are saved in FITS binary table extensions and opened as astropy.table.Table objects. Cubes/images/spectra are saved in dictionaries of mpdaf.obj objects.

```
In [4]: f, ax = plt.subplots(2,3)
In [5]: src.images['MUSE_WHITE_LYANA'].plot(scale='arcsinh', ax=ax[0,0],title=r'white')
In [6]: src.images['MUSE_LYA_LYANA'].plot(scale='arcsinh', ax=ax[0,1], title='NB'+r'$ly\alpha$')
In [7]: src.images['MUSE_CONT_LYANA'].plot(scale='arcsinh', ax=ax[0,2], title='continuum')
In [8]: src.images['MUSE_CONT_MASKED_LYANA'].plot(scale='arcsinh', ax=ax[1,0], title='mask')
In [9]: src.images['HST_F814_LYANA'].plot(scale='arcsinh',zscale=1,ax=ax[1,1],title='HST F814W')
In [10]: src.images['MUSE_PSF_LYANA'].plot(scale='log', ax=ax[1,2], title='PSF')
```



Advanced reduction tasks

- Handle the pixel table created by the data reduction system
- Extract a subset of a pixtable using spatial, spectral or origin (IFU, slice, CCD)
- Estimate a reference sky spectrum
- Compute and apply a flat field correction to the slice level
- Mask and remove instrumental artifacts
- Combine several cubes



References: Bacon & al., 2015, A&A, 575, A75.1411.7667
Conseil & al, 2017, ADASS XXVI

mpdaf
muse python data analysis framework

https://git-cral.univ-lyon1.fr/MUSE/mpdaf

mpdaf-support@osulistes.univ-lyon1.fr

Laure Piqueras

laure.piqueras@univ-lyon1.fr

CRAL CENTRE DE RECHERCHE ASTROPHYSIQUE DE LYON
muse multi unit spectroscopic explorer
erc European Research Council
cnrs dépasser les frontières