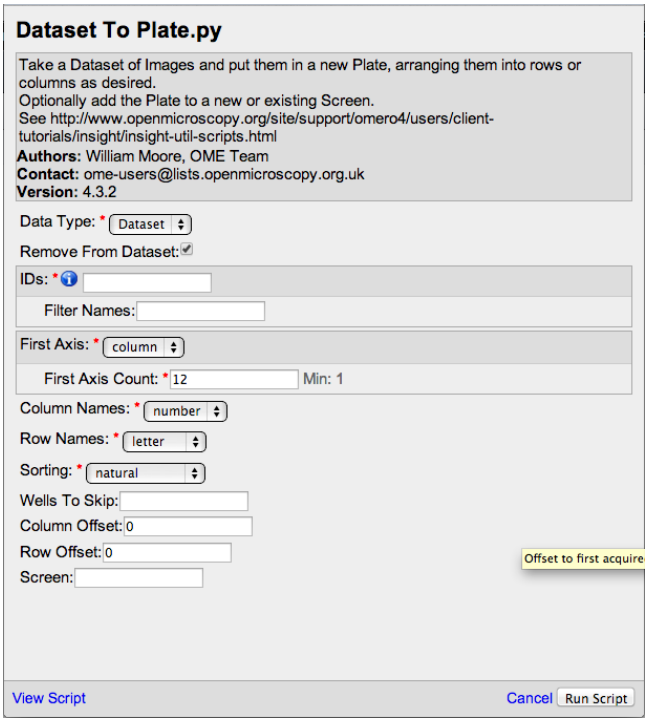
Dataset_to_plate.py script changes.

Changes added to the script allow for more flexible placement of wells on the plate when converting dataset to a plate.



List of changes:

- Row and column offset. If acquired data does not start at column 1 or row 1, images can be offset to start at arbitrary row or column.
- Well skipping. Allows to leave empty wells if data for a given well is not provided.
- Natural sorting. Option added to place images in correct wells for different naming conventions.

When data available does not start at row 1 and/or column 1 user can input offset values to **Row Offset** and **Column Offset** edit boxes which will leave Offset number of rows and/or columns empty.

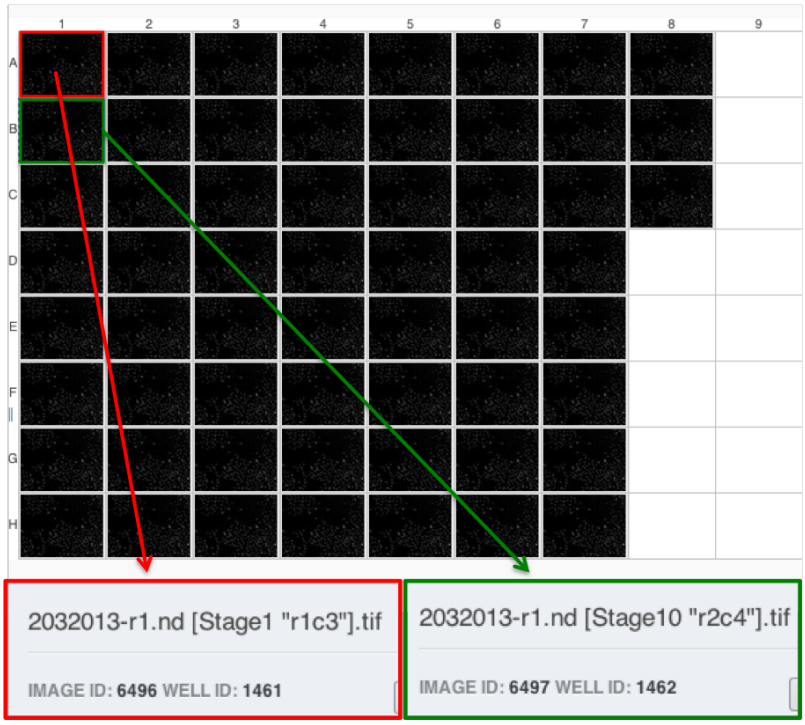
When not all of the images are present and some wells should be left empty user can input list of wells to skip to **Wells To Skip** edit box in following format:
Row:Column,Row:Column,Row:Column, ….

If images are named using natural naming convention: Image 1, Image 2, … , Image 10, … , natural sorting option should be selected from **Sorting** combo box as Alphabetical sorting will result in incorrect image order: Image 1, Image 10, …. , Image 2, … .

Example Use.
Example dataset can be found at:
https://github.com/emilroz/ExampleData/tree/master/Dataset_to_Plate.py This data set was created to test new functionality.

1. Running script without new functionalities for this dataset results in wrong plate layout.
Settings (emulate previous version of the script):

- First Axis: Row (images are labeled row:column)
- Axis Count: 8 (highest row number)
- Sorting: alphanumeric
- Wells to skip:
- Column offset: 0
- Row offset: 0



Alphabetical sorting used in Dataset_to_plate.py will result in wrong placment of images (listed on the left) on the plate (see below - images placed on plate along rows).

2. Running script with correct settings.

Settings:

- First Axis: Row (images are labeled row:column)
- Axis Count: 8 (highest row number)
- Sorting: natural (this will properly sort images 1,2,3,4,5,6,7,8,9,10,….)
- Wells to skip: (format rowA:columnA,rowB:colB, …)
  1:10,2:10,3:10,4:10,5:10,6:10,1:11,2:11,3:11,4:11,5:11,6:11,7:11
- Column offset: 2 (data starts at column 3)
- Row offset: 0