Planet Orders API Workshop

The new Planet API ordering service, Orders API, makes it easier to create pipelines to continuously ingest imagery for processing and analysis. Attendees will be introduced to new tools, functionality, and product types available in this service.

Terms and Definitions

- **Item**: An entry in our catalog, and generally represents a single logical observation (or scene) captured by a satellite
- **Item Type**: Represents the class of spacecraft and/or processing level of an item
- **Asset**: A product that can be derived from an item's source data, and can be used for various analytic, visual or other purposes
- **Bundle**: A group of Assets for an Item and associated metadata
- **Order**: A single request for a subset of Planet imagery, comprised of bundles
- **Tool**: An optional pre-processing algorithm applied to the contents of an Order

Catalog Documentation: https://developers.planet.com/docs/api/items-assets/
Bundle Reference: https://developers.planet.com/docs/orders/product-bundles-reference/

API Mechanics

Base Endpoint: `https://api.planet.com/compute/ops/orders/v2`
Authorization Header: "Authorization: api-key xxxx…"
Alternative Authorization: Basic-Auth with api-key as username.
Install Python CLI: "pip install planet --user"
Initialize Python CLI: "planet init" and login with your provided credentials

Anatomy of an Order

Orders Documentation: https://developers.planet.com/docs/orders/

```
{
    "name": "My Example Order",
    "products": [
        {
        "item_type": "PSScene4Band",
        "product_bundle": "analytic",
        "items": ["20191003_183407_1032", "20191003_183408_1032", ...]
        }
    ],
    "notifications": {
        "email": true
    },
    "delivery": {
        "archive_type": "zip",
        "single_archive": true,
        "archive_filename": "{{order_id}}_{{name}}.zip"
    },
    "tools": [
        {
            "clip": {
                "aoi": ...GeoJSON...
            }
        }
    ]
}
```

- **name**: Freeform identifier or description of the order
- **products**: The list of Product Bundles to be ordered
- **notification**: Configuration for email or webhooks
- **delivery**: Configuration for delivery format and cloud publishing
- **tools**: A collection of operations to apply over each ordered Product Bundle

Common Tools

Documentation: https://developers.planet.com/docs/orders/tools-reference/

Clip

Clip takes a GeoJSON Polygon Or MultiPolygon AOI as it's argument. The Clip operation excludes all pixels outside of the provided AOI from processing.

Arguments:

- **aoi:** GeoJSON Polygon/MultiPolygon up to 500 vertices

```
{ "clip": {
           "aoi": {
                "type":"Polygon",
                "coordinates": [
                     [
                          [-163.828125, -44.59046718130883],
                          [181.7578125, -44.59046718130883],
                          [181.7578125, 78.42019327591201],
                          [-163.828125, 78.42019327591201],
                          [-163.828125, -44.59046718130883]
                     ]
                ]
           }
} }
```

Bandmath

Applies arbitrary bandmath to produce derived output

Arguments:

- **b1 / b2 / b3 / b4 / b5:** A math expression defining how the output band should be computed.
- **pixel_type:** Determines the data type of the output pixel.
    - One of "8U" (8bit unsigned), "16U" (16bit unsigned), "16S" (16bit signed), "32R" (32bit floating point) or "Auto" (determined based on source)

```
{ "bandmath": {
      "b1": "(b4 - b3) / (b4 + b3)",
      "pixel_type": "32R",
} }
```

Reproject

Reproject, resample and rescale raster products to a new projected coordinate system, and resolution.

Arguments:

- **projection**: The new coordinate system
  - EX: EPSG:4326 for WGS84, EPSG:32611 for UTM 11 North (WGS84), or EPSG:3857 for Web Mercator
- **resolution**: The width and height of pixels in the output projection system
- **Kernel**: Selects which resampling kernel to use.
  - One of "near", "bilinear", "cubic", "cubicspline", "lanczos", "average" and "mode"
  - Further details: https://gdal.org/programs/gdalwarp.html#cmdoption-gdalwarp-r

```
{ "reproject": {
  "kernel": "cubic",
  "projection": "EPSG:4326"
} }
```

Composite

Combines a set of raster images into a single image. This tool has no arguments, but note that images must all be in the same projection. So, this tool is usually combined with the Reproject tool.

```
{ "composite": {} }
```

Demo 1

Simple HTTP request via cURL:

```
curl -s -H 'Content-Type: application/json' \
      -u <Planet API key>: \
      https://api.planet.com/compute/ops/orders/v2 \
      -d '{ "name": "Clip demo", "products": [ { "item_ids": [
"20191009_212140_0f4c", "20191009_212139_0f4c" ], "item_type": "PSScene4Band",
"product_bundle": "analytic" } ], "tools": [ { "clip": { "aoi": { "coordinates":
[ [ [ -121.5633773803711, 38.252740356338236 ], [ -121.52870178222656,
38.25381877635746 ], [ -121.50123596191405, 38.30771936971289 ], [
-121.52114868164061, 38.30260053372373 ], [ -121.5757369995117,
38.310143955413096 ], [ -121.58912658691406, 38.3225350164105 ], [
-121.59667968749999, 38.277539967611396 ], [ -121.5633773803711,
38.252740356338236 ] ] ], "type": "Polygon" } } } ] }'
```

You will need to substitute `<Planet API key>` above with your assigned key.

Demo 2

You will run this command:

```
planet orders create
    --name 'A more complex order'
    --id 20191009_212140_0f4c,20191009_212139_0f4c
    --bundle analytic
    --item-type PSScene4Band
    --zip bundle
    --tools tools.json
    --cloudconfig s3.json
```

First, save the following as **tools.json**:

```
[
  {
    "clip": {
      "aoi": {
        "type": "Polygon",
        "coordinates": [
          [
            [-121.5633773803711,38.252740356338236],
            [-121.52870178222656, 38.25381877635746],
            [-121.50123596191405, 38.30771936971289],
            [-121.52114868164061, 38.30260053372373],
            [-121.5757369995117, 38.310143955413096],
            [-121.58912658691406, 38.3225350164105],
            [-121.59667968749999, 38.277539967611396],
            [-121.5633773803711, 38.252740356338236]
          ]
        ]
      }
    }
  },
  {
    "reproject": {
      "kernel": "cubic",
      "projection": "EPSG:4326"
    }
  },
  {
    "composite": {}
  }
]
```

Modify, then save the following as **s3.json** (optional; if you don’t have S3 creds, just remove **`--cloudconfig s3.json`** from the command above.)

```
{
   "amazon_s3": {
     "bucket": "<YOUR BUCKET>",
     "aws_region": "<AWS REGION>",
     "aws_secret_access_key": "<REDACTED>",
     "aws_access_key_id": "<REDACTED>",
     "path_prefix": "2019-planet-explore-conference/"
   }
 }
```