Overview of Foundry Assets for Data Hub

Mapping of EDAV **Databricks** asssets to **1CDP** Foundry

Code App User Interfaces

1CDP Code App Options

- Code Repositories "**Advanced pipelines**"
  - Suggested: For Data Hub Engineering, not analyst use, with **Pipeline Builder** and **Data Lineage**
  - "Data Pipeline Management"
  - **Governance**: "Prioritizes change traceability and governance to ensure that critical pipelines remain secure and robust; advanced review and approval workflows and complete changelogs."
  - Git backed
  - Spark with advanced compute configuration
  - Ontology Support
  - Python, SQL, Java, Mesa
  - Console Support with Debug Mode
  - Visualization Support
  - Supports Foundry data management libraries and publishing custom Python libraries
  - Security
  - Advanced Workflow Review
  - Unit Testing
  - Define, Maintain, and Publish Custom Python Libraries
- Code Workspaces "**Exploratory Analysis**"
  - Suggested: For Analyst Use on "Gold" Medallion Data, or Transform Silver to Gold Medallion
  - "Data Exploration Management"
  - **Flexibility**: "Prioritizes rapid and flexible iteration with full branching support and automatic Git versioning"
  - Kubernetes only, no Spark
  - No Ontology Support
  - Python, R (no SQL)
  - RStudio or Jupyter Notebooks
  - Consume pip, CRAN, conda and Code Repository libraries
  - Git backed
  - Interactive EDA
- Code Workbook "**Advanced Analysis**"
  - Suggested: For Final Analysis, Ad Hoc EDA, SQL
  - "**Rapid Changes**": "Prioritizes rapid iteration and collaboration with a lightweight branching workflow; does not require CI checks or unit testing"
  - Spark (warm)
  - Ontology Support
  - SQL, Python, R
  - Visualization
  - not git backed, lightweight branching
  - Interactive EDA

1CDP Artifacts (Package Repository and Management)

- Curated Package Management
  - pypi,maven,conda,docker,npm
  - no R/CRAN
  - custom packages
  - Multiple artifact storage for versioning, curation
- Upload via API with timed Token

```
export TOKEN=<TOKEN>
export PACKAGE=package-1.0.0-py36_0.tar.bz2 # replace with your package's local
path
export PLATFORM="linux-64" # replace with the package platform (e.g. noarch,
linux-64, osx-64)
curl -H "Authorization: Bearer $TOKEN" -H "Content-Type: application/octet-stream"
--data-binary "@$PACKAGE" -XPUT
"https://1cdp.cdc.gov/artifacts/api/repositories/ri.artifacts.main.repository.cfec
472f-841a-463c-aeba-48fc5b187c32/contents/release/conda/$PLATFORM/$PACKAGE"
```