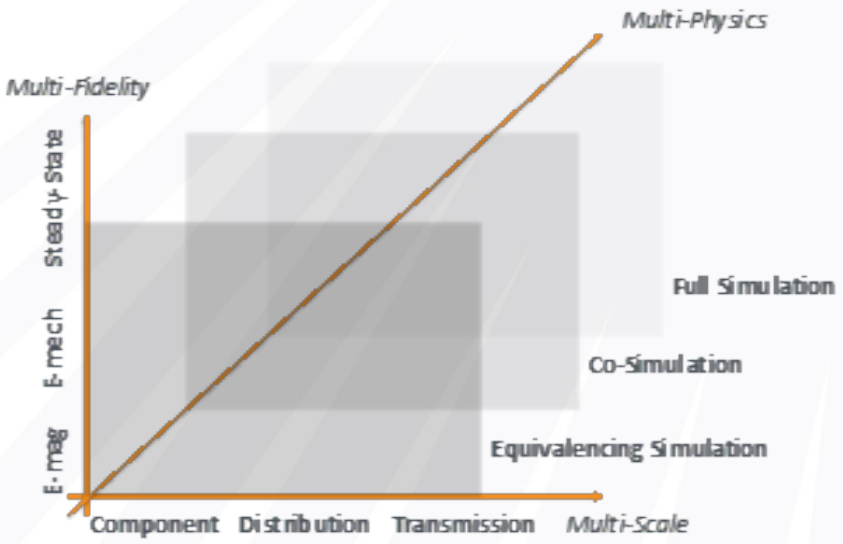
Lab Call – FY22 Core Topic – OEDI SI

- Main goals
  - Provide access to data, data integration & mapping information/scripts
  - Physics-/network- & data-/ML-based solar analytics
  - Enable reproducible, robust, replicable and generalizable R&D in simulation and emulation of solar system integration
  - Encourage/enforce dataset format/I/O standardization
- National Labs collaboration
  - ANL, NREL, ORNL, PNNL
  - 2022 Oct. to 2024 Sept.
- Main deliverables
  - Generic and integrated datasets from four power systems operation technologies
    - smart inverters/distributed energy resources management systems (DERMS)
    - phasor measurement units (PMUs) and data concentrators (DCs)
    - smart meters/advanced metering infrastructure (AMI)
    - new sensor technologies
  - Benchmark analytics (physics and ML based algorithms)
    - Distribution state estimation
    - Optimization functions
    - Transients' analysis
    - Data interfaces (CIM, OpenDSS, Ditto, more in the future

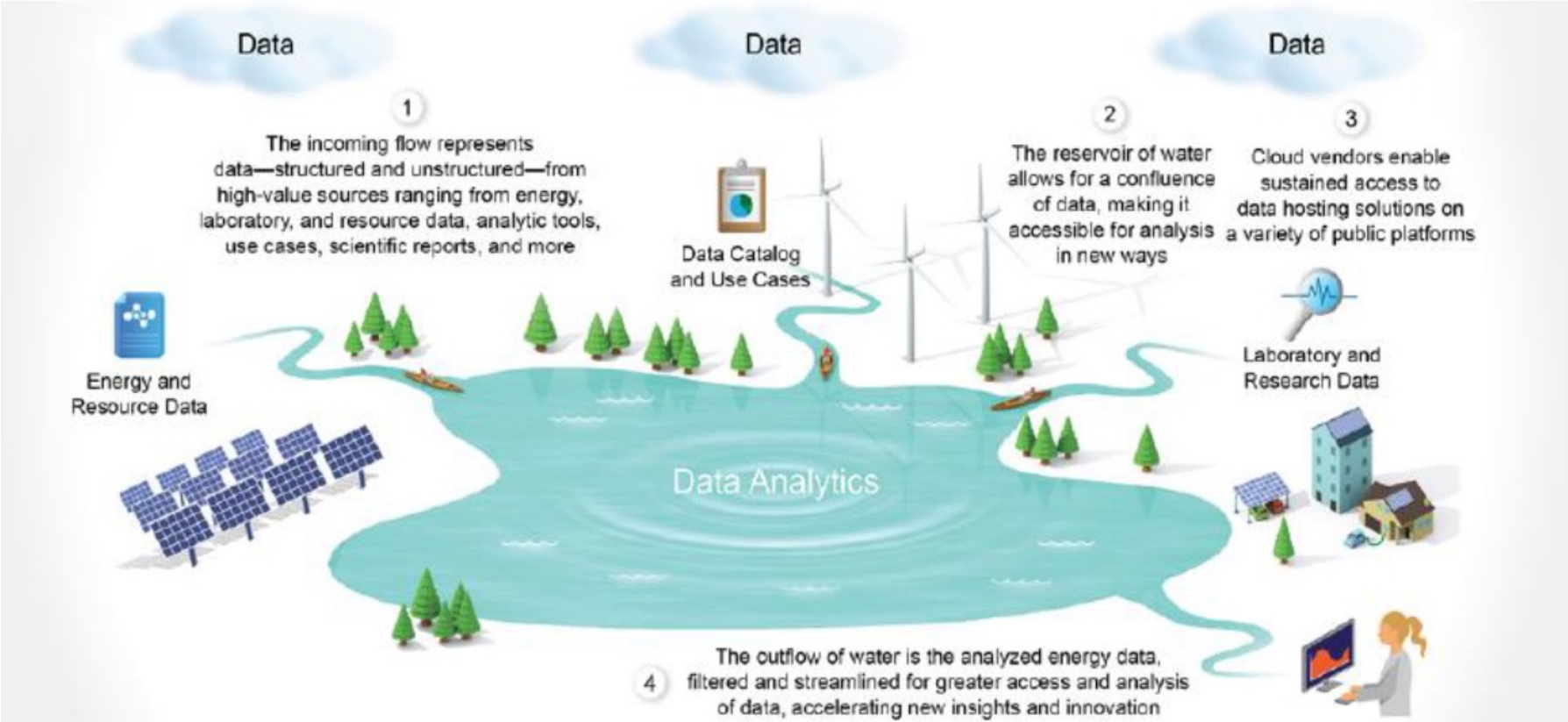
Challenges in Data Integration, Modeling & Simulations

		Data	
		Same	Different
Analysis	Same	Reproducible	Replicable
	Different	Robust	Generalisable

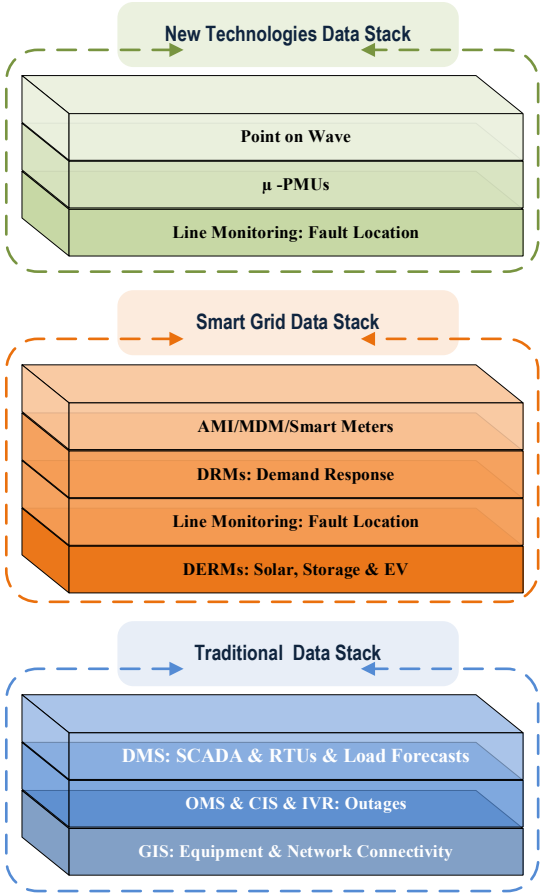
Whitaker (2018) https://doi.org/10.6084/m9.figshare.7140050.v2

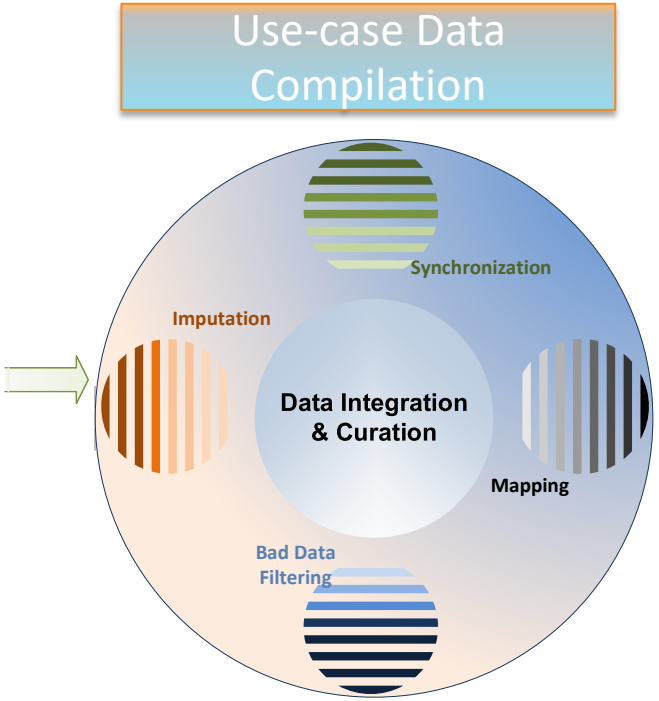


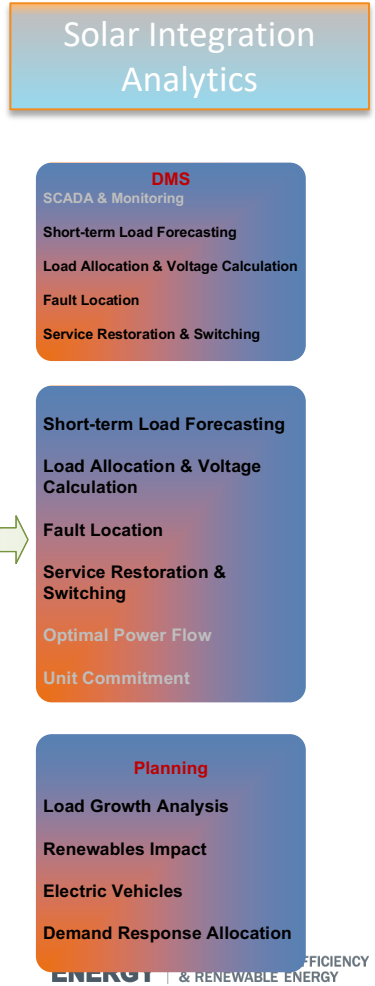
Open Energy Data Initiative (OEDI) – Data Lakes



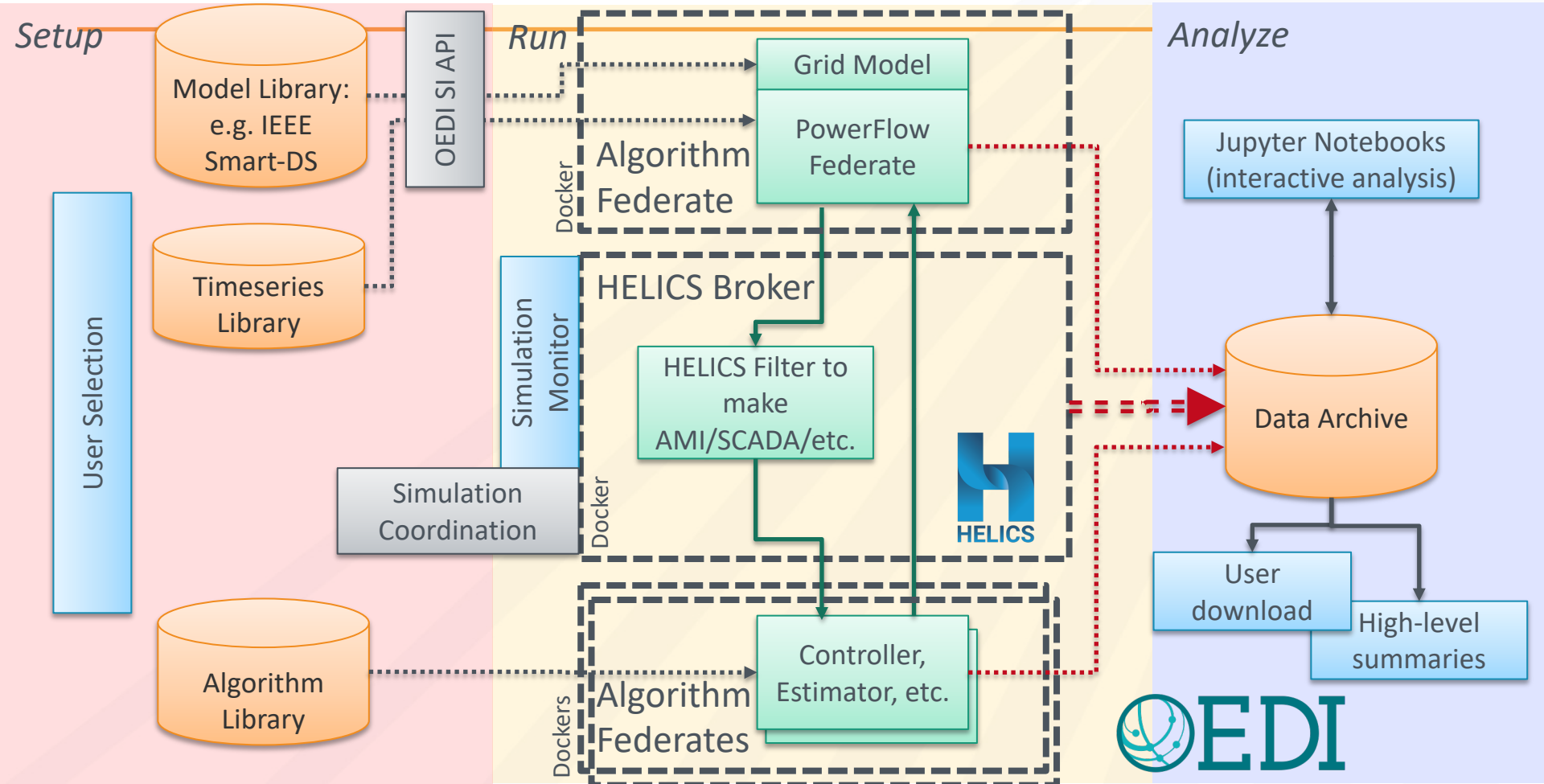
OEDI SI


New Technologies Data Stack
Point on Wave
µ -PMUs
Line Monitoring: Fault Location
Smart Grid Data Stack
AMI/MDM/Smart Meters
DRMs: Demand Response
Line Monitoring: Fault Location
DERMs: Solar, Storage & EV
Traditional Data Stack
DMS: SCADA & RTUs & Load Forecasts
OMS & CIS & IVR: Outages
GIS: Equipment & Network Connectivity

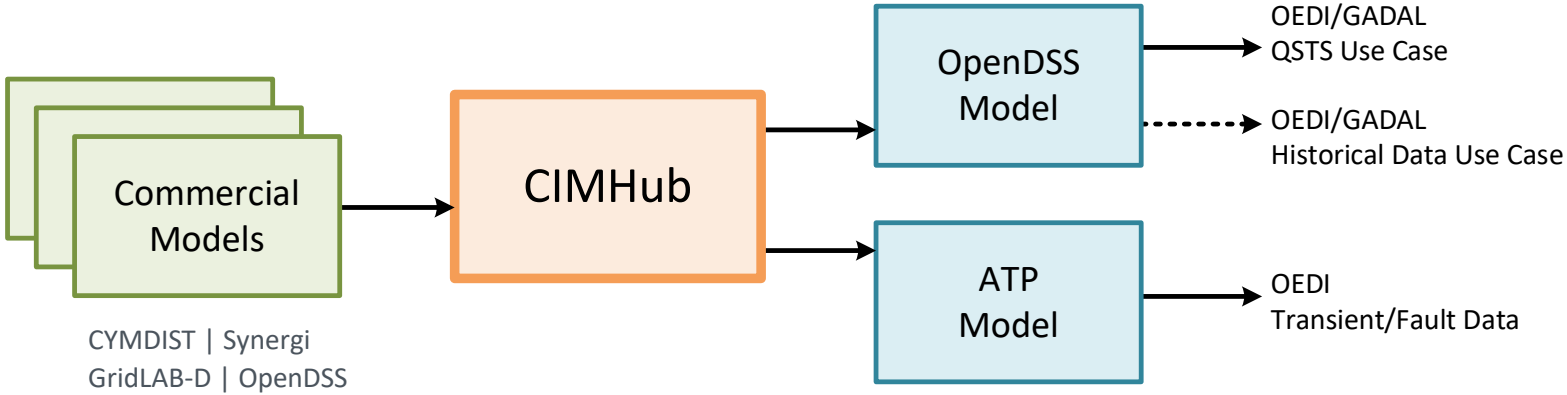

Use-case Data Compilation
Synchronization
Imputation
Data Integration & Curation
Mapping
Bad Data Filtering


Solar Integration Analytics
DMS
SCADA & Monitoring
Short-term Load Forecasting
Load Allocation & Voltage Calculation
Fault Location
Service Restoration & Switching
Short-term Load Forecasting
Load Allocation & Voltage Calculation
Fault Location
Service Restoration & Switching
Optimal Power Flow
Unit Commitment
Planning
Load Growth Analysis
Renewables Impact
Electric Vehicles
Demand Response Allocation

Conceptual Diagram [Note: Use-case based (In addition to Lake)]



OEDI SI API: CIMHub converts models to OpenDSS & ATP



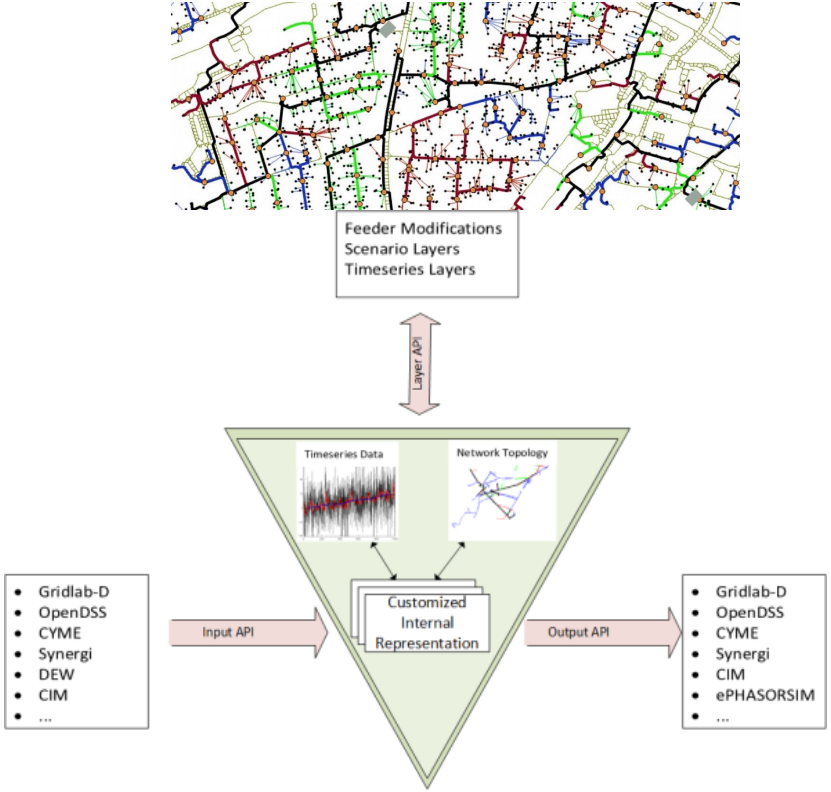
https://github.com/GRIDAPPSD/CIMHub/tree/feature/SETO

CIMHub Docker-free Option:

- Java JDK 11+
- download Blazegraph & CIMHub JAR files
- pip install cimhub

OEDI SI API: Distribution Transformation Tool (DiTTo)

- Feeder Conversion
- Comprehensive Feeder Metric Computations
- Graph Theory Network Analysis
- Feeder Modification:
  - Modify loads
  - Add solar
  - Set controls of components
  - Add/remove electrical components
  - And more!



Early Version Tester

- Expectations
  - Test the current environment
  - Discuss what they can bring from their projects to OEDI – SI
    - Data
      - Public
      - Private (Confidential)
    - Models
    - Algorithms
  - Become founding members in the OEDI – SI Users Group
- Each testers will be assigned an OEDI – SI SME
  - A follow-up meeting will be setup with each tester

Funded by:
SOLAR ENERGY TECHNOLOGIES OFFICE
U.S. Department Of Energy

Setting Up Docker Environment

- Docker container
  - Standard unit of software that packages up code and all its dependencies
  - Replicate results of a workflow on different systems without having to load all the different dependencies that each component may require

Install Docker on Ubuntu

To install docker on a linux environment, firstly update the linux package manager and install dependencies to allow the package manager to install them over https:

```
$ sudo apt-get update
$ sudo apt-get install ca-certificates curl gnupg lsb-release
```

Then add docker's GPG keys and set up the repository with the commands:

```
$ sudo mkdir -p /etc/apt/keyrings
$ curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo gpg --dearmor -o /etc/apt/keyrings/docker.gpg
$ sudo mkdir -p /etc/apt/keyrings
$ curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo gpg --dearmor -o /etc/apt/keyrings/docker.gpg
```

Finally update the package index and install the latest version of docker:

```
$ sudo apt-get update
$ sudo apt-get install docker-ce docker-ce-cli containerd.io docker-compose-plugin
```

Install Docker Desktop on Windows

- Download the installation .exe
  https://docs.docker.com/desktop/install/windows-install/
- Run the installer
- Follow the instructions on the installation wizard
- Detailed instructions in the User Guide

Funded by:
SOLAR ENERGY TECHNOLOGIES OFFICE
U.S. Department Of Energy

Running DSSE Docker Container(s)

Using NREL's DSSE container as an example

- Load Docker Image

```
$ docker load < example.tar.gz
```

- Connect Volume
  - Linux

```
$ mkdir outputs_build

$ docker volume create --name gadal_output --opt type=none --opt
device=/path/to/folder/outputs_build --opt o=bind
```

  - Windows

```
$ docker volume create --name gadal_output --opt type=none --opt
device=/c/path/to/folder/outputs_build --opt o=bind
```

- Run the Container

```
$ docker run --rm --mount source=gadal_output,target=/simulation/outputs gadal-
example:0.0.0
```

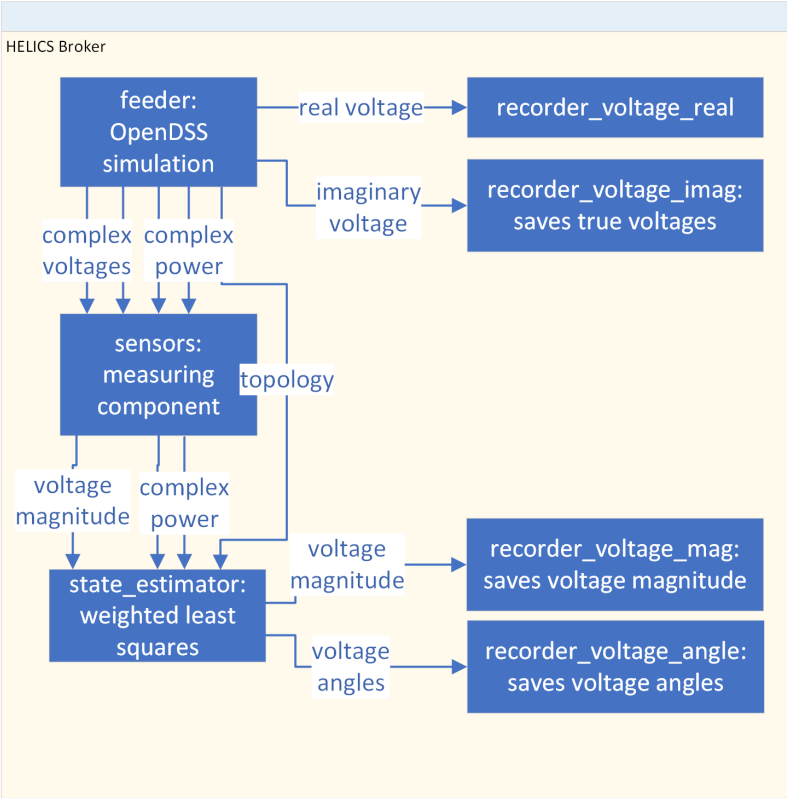
Please refer to the README in each container for detailed instructions

Funded by:
SOLAR ENERGY TECHNOLOGIES OFFICE
U.S. Department Of Energy

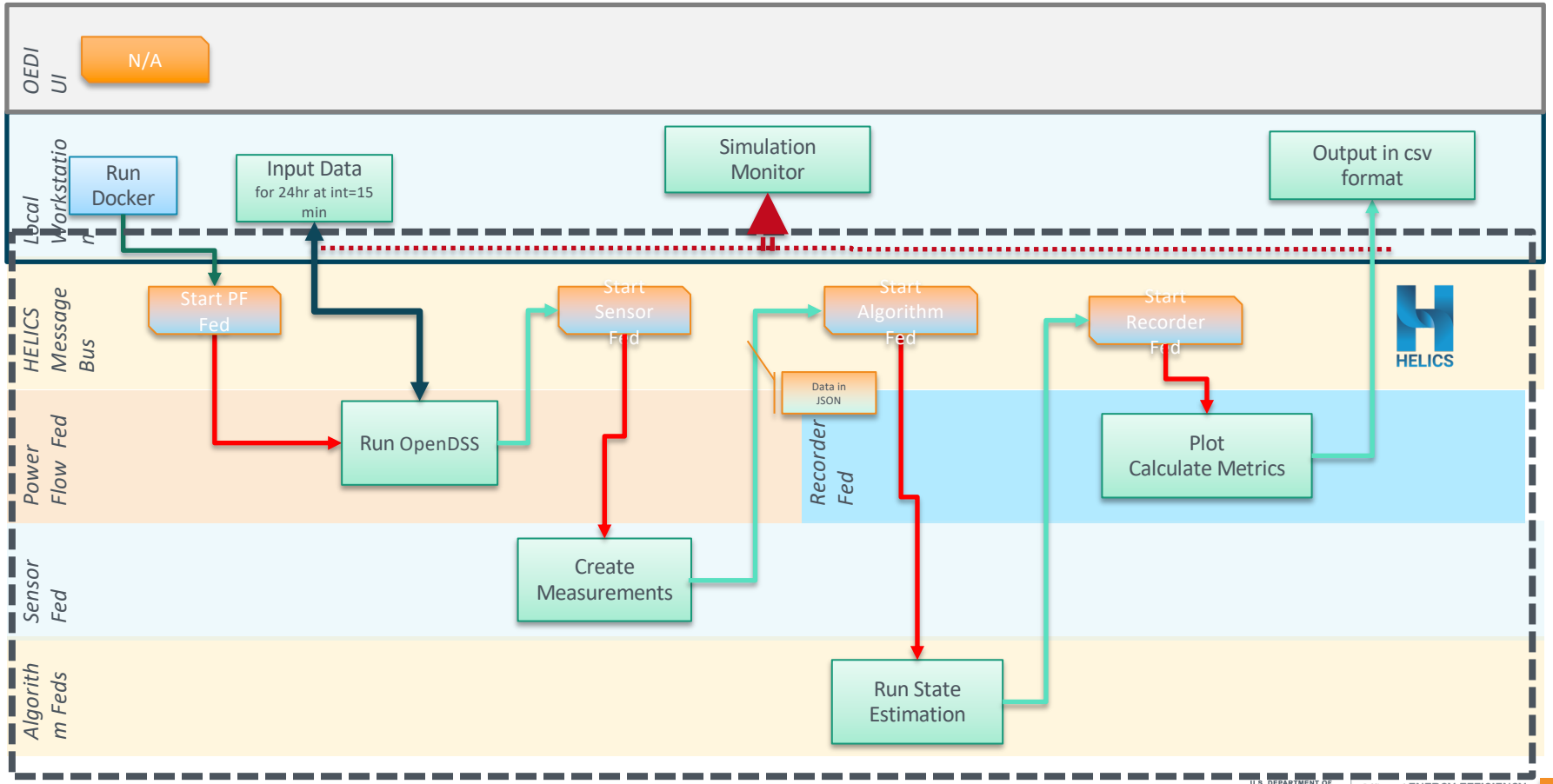
DSSE Use-case Example

Components

- OpenDSS feeder simulation
- Basic measurement federate
- Recording federates
- Distribution system state estimator
- Plotting script to visualize the results



DSSE Use-case Example



Docker Container