EDC Hackathon II: Building a Federated Dataspace Catalog

The Eclipse Dataspace Connector Project

- Key Challenges with Federated Catalogs
- EDC Federated Catalog Architecture and Topologies
- Tasks

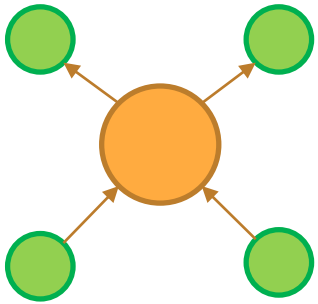
Federated Catalog: The Challenge

- How do participants advertise data at-scale in a dataspace?
  - Potentially thousands of global participants (e.g., complex supply chains)
  - Data comes in all shapes and sizes: big data, streams, data services (e.g., APIs)
  - Data must be described and have transparent usage policies
  - Data must be instantly searchable
- Key technical problem revolves around the fact that in most cases, not all data in a dataspace is public
  - Some data requires data consumers to adhere to usage requirements
  - Some data may have access restrictions based on an identity or verifiable credentials
  - Identities may be defined in multiple jurisdictions

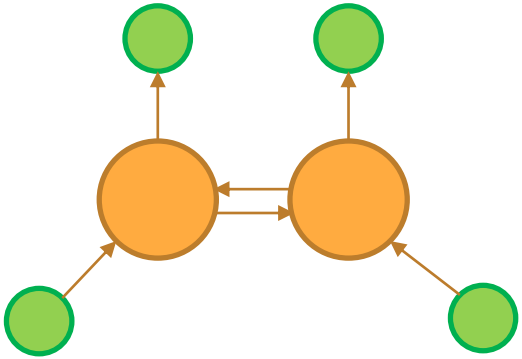
ECLIPSE
INCUBATION

Fully- and Semi- Centralized Catalog Architectures

- Require a broker where participants publish their catalogs



Centralized Broker Dataspace



Semi-Centralized Dataspace

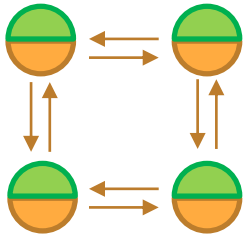
ECLIPSE
INCUBATION

Centralized Catalog Architectures Common Issues

- Data Visibility and Sovereignty
  - Is it acceptable for a third-party to have access to an organization’s data catalog?
  - Is it acceptable for an organization to rely on a third-party to advertise its data?
  - Can a third-party catalog provider properly enforce an organization's access rules?
- Reliability and Scalability
  - In fully-centralized systems, what happens when the catalog is down?
  - In semi-centralized systems, how can replication-at-scale be managed?

The EDC Federated Dataspace Catalog

- Solves the problems of data visibility and enterprise scalability &reliability
- Implements a crawler architecture
  - Each node consists of a Federated Cache Node **(FCN)** and a Federated Cache Crawler **(FCC)**
  - The FCN makes its asset catalog available to other participants
  - The FCC crawls other FCN instances on a periodic basis and caches the results

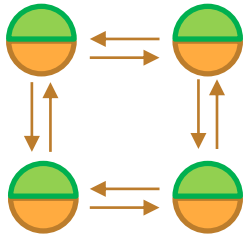


ECLIPSE
INCUBATION

The Issue of Data Visibility

- FCC presents its identity and credentials to an FCN
- The FCN uses the same modules as the Connector to run policy access and usage control checks to filter the returned assets
  - Organizations maintain control of their asset catalogs and access control
  - Through extensibility, organizations may also implement custom access control logic

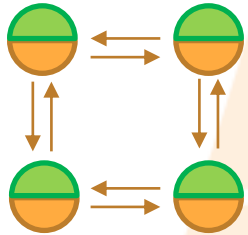
Exchange and validation of credentials at each point



ECLIPSE
INCUBATION

Scalability and Reliability

- Each FCC node caches its results
- Enables instantaneous distributed queries since asset catalogs are mirrored locally
    - Only assets the client node is entitled to view
- The dataspace becomes fault tolerant and resilient
    - If the origin FCN is down, the local cached copied will continue to work



ECLIPSE
INCUBATION

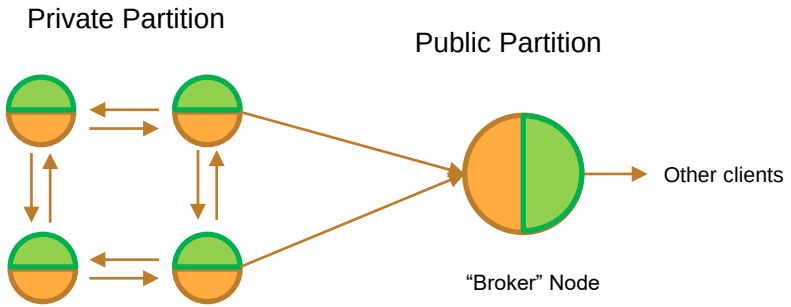
Architectural Highlights

- Support for multiple identity providers
  - For the Hackathon we have enabled distributed identities using Web DIDs
  - The same catalog architecture works without change using OAuth2, Web DIDs, ION/Blockchain
  - May potentially use a combination of all of the above
- Pluggable backend catalog sources
  - Use your favorite backing catalog system, e.g., GXFS, Apache Atlas, a database, etc.
- Can integrate custom query languages
- Built on the same foundation as EDC

ECLIPSE
INCUBATION

Deployments

- The FCN and FCC made be deployed in a connector process or as separate services (recommended)
- The crawler architecture is designed for peer-to-peer but can also support broker models or a hybrid combination
    - For example, a dataspace may have private data shared via a peer-to-peer partition and public data offered via a broker



ECLIPSE
INCUBATION

Hackathon Tasks

Hackathon Tasks

- Integrate a backing catalog system
  - Involves create a data seeder to make assets available in the dataspace
- Custom access control via a credential verifier
  - Implement custom catalog access control on top of the distributed identity (Web DID) system
- Expose a data service as a catalog asset (based on Amadeus use case)
- User Interface
  - Build a UI to query and display the dataspace catalog
- Cloud deployment
  - Deploy the catalog nodes to your favorite cloud environment and host Web DIDs
- Implement a custom query language for the catalog

ECLIPSE
INCUBATION