LF AI & DATA

The EGERIA Metadata Show

Egeria Webinar

CONNECTOR COMPARISON

SHOULD YOU USE AND INTEGRATION CONNECTOR OR REPOSITORY CONNECTORS TO INTEGRATE A DATA CATALOG INTO THE OPEN METADATA ECOSYSTEM?

Mandy Chessell CBE FREng
Egeria Open Source Project Lead

Egeria's webinar series

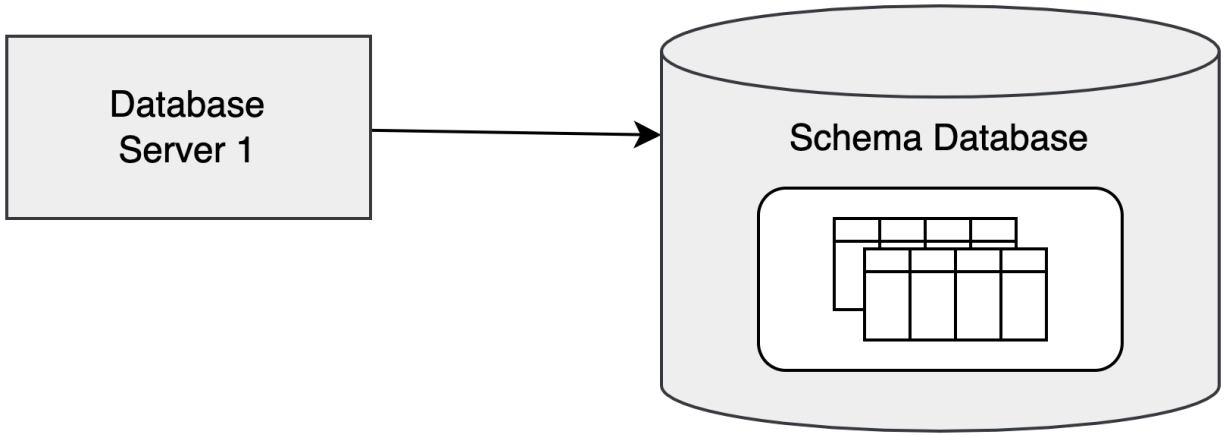
~~7th~~ **15th March 2022**	~~15:00~~ 14:00 UTC	**How to build an integration connector**	This session covers how to extend Egeria's automated cataloguing to include metadata from a new technology. It describes how automated cataloguing works and the role of the integration connector. It covers the design of the integration connector using examples to illustrate the different approaches and their benefits and and challenges. It shows how to set up a project for a new connector, how to build and package it and finally it shows the new connector running in Egeria. Zoom Conference https://zoom.us/j/523629111	**Mandy Chessell**
4th April 2022	15:00 UTC	**How to choose: Integration or repository connector?**	This session compares using Integration Connectors with Repository Connectors to connect technologies into Egeria. We will go through the pros and cons of integration connectors and both types of repository connectors (native and proxy) and how these choices impact and benefit your Egeria eco-system. Zoom Conference https://zoom.us/j/523629111	**Mandy Chessell**
9th May 2022	15:00 UTC	**Using a repository connector**	This session covers how to use Repository Connectors to connect technologies into Egeria; focussing on XTDB (formerly known as crux). Ever wanted to know what the state of your metadata was at some specific time in the past? This session will introduce the XTDB open metadata repository that supports these historical metadata queries. Zoom Conference https://zoom.us/j/523629111	**Chris Grote**
6th June 2022	15:00 UTC	**Kubenetes operators and Egeria**	This session will cover how easy it is to run Egeria in Kubenetes and how the Egeria Kubenetes operator can be used to manage Egeria in a Kubenetes environment. Zoom Conference https://zoom.us/j/523629111	**Nigel Jones**

Agenda

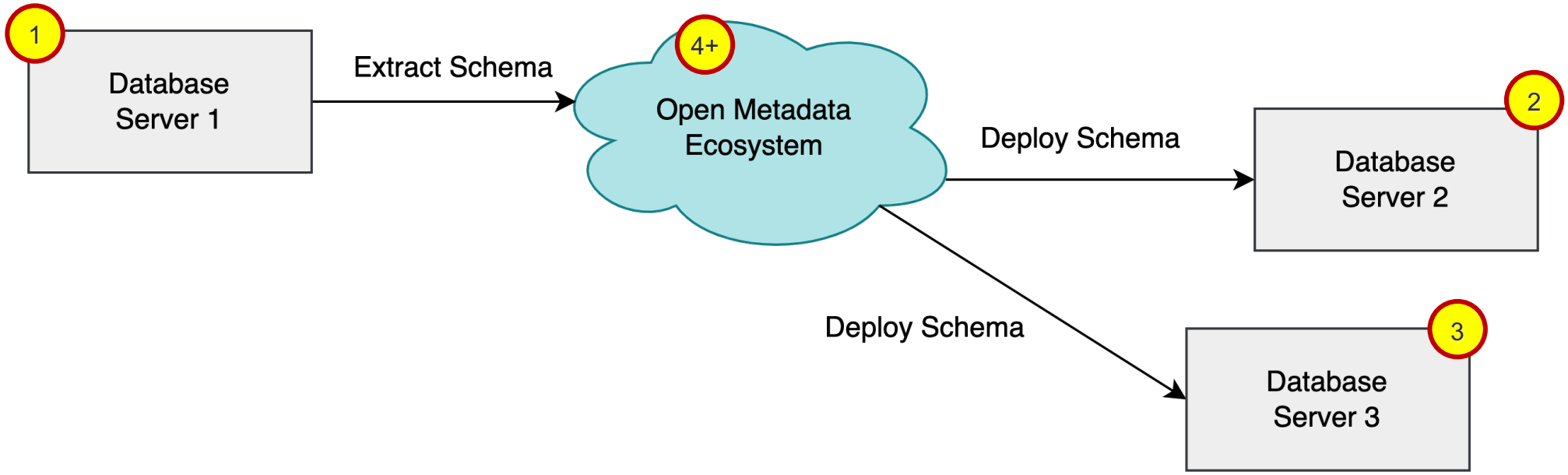
- Background to metadata and data catalogs
- Coco Pharmaceuticals use case
- Connector comparison
  - Infrastructure comparison
  - Federated queries
  - API comparisons
- Conclusion

Metadata and Data Catalogs

The metadata collection

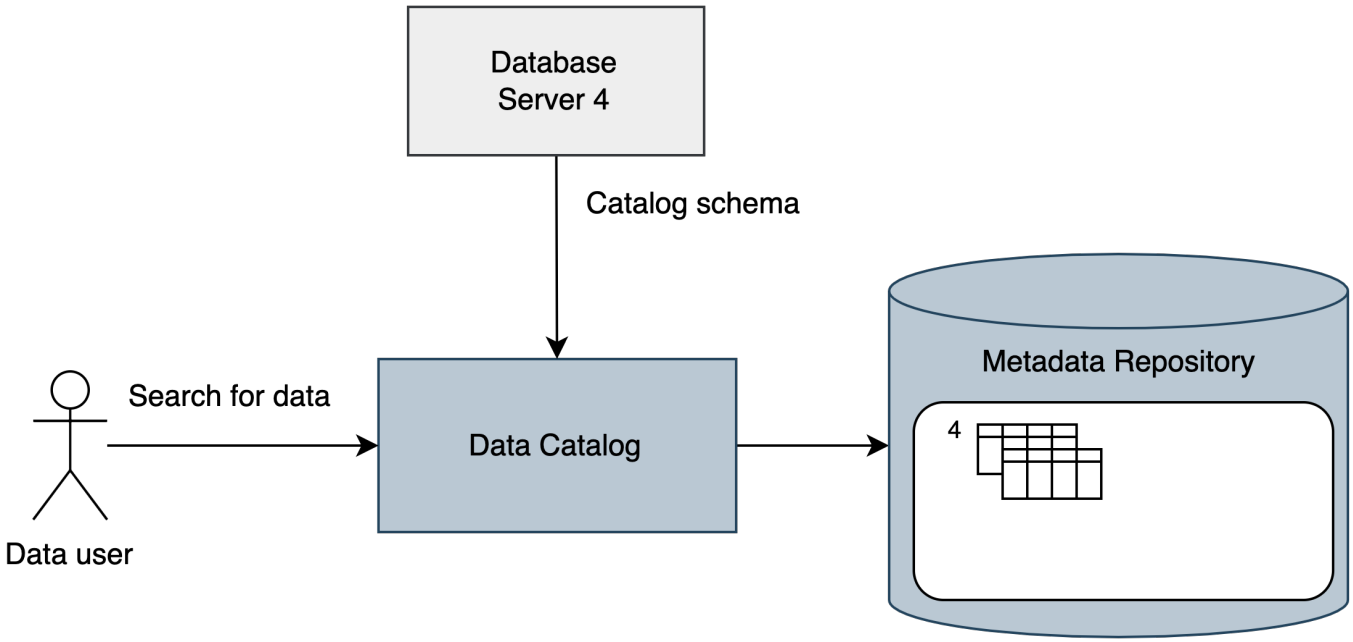


How many metadata collections?



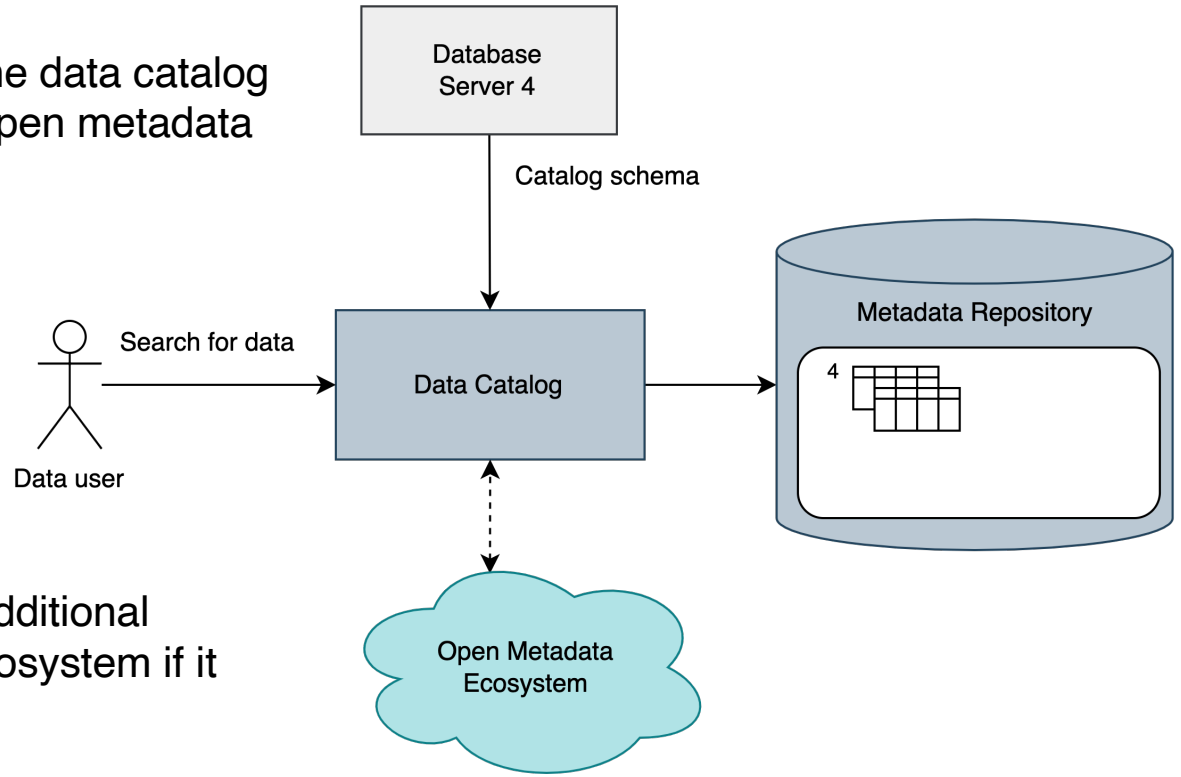
Using a data catalog

1. Search
2. Augmentation



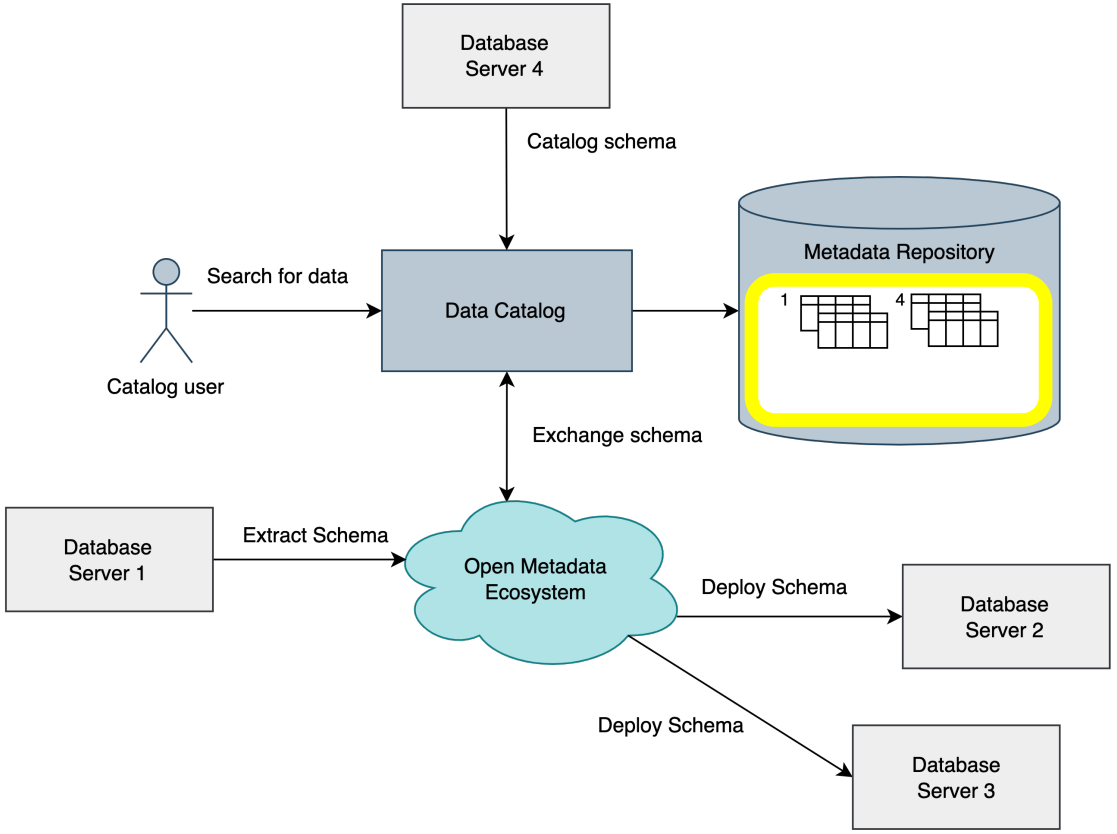
Why connect to the open metadata ecosystem?

Database Server 4 is not visible to the open metadata ecosystem until the data catalog is exchanging its metadata with the open metadata ecosystem.



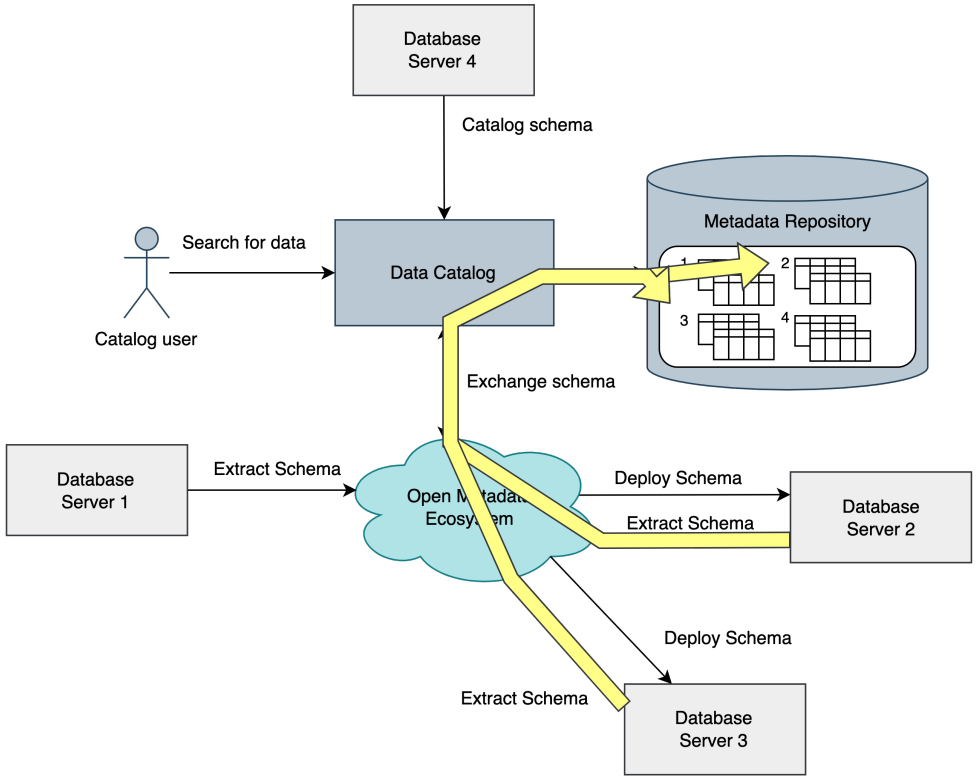
The data catalog can get access to additional metadata from the open metadata ecosystem if it connects

Getting access to more metadata …



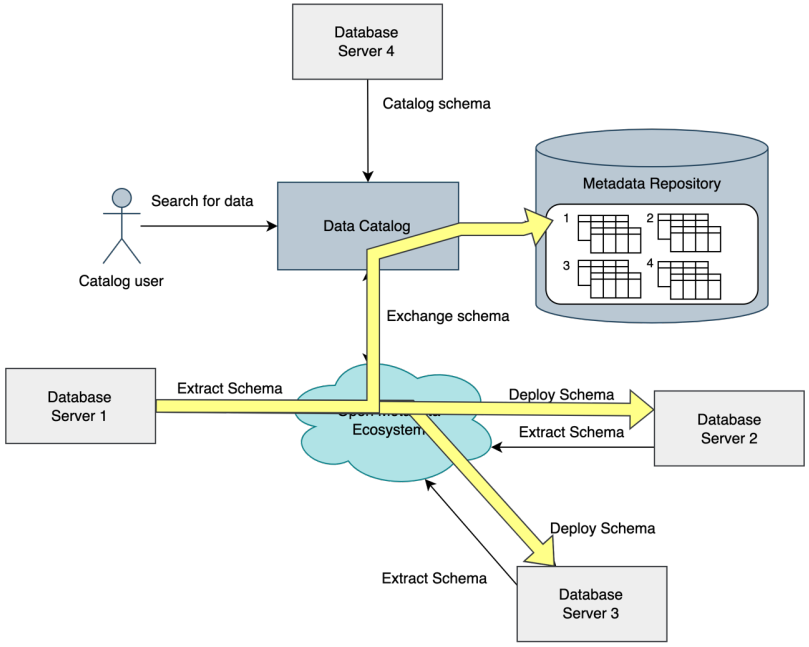
Getting access to more metadata …

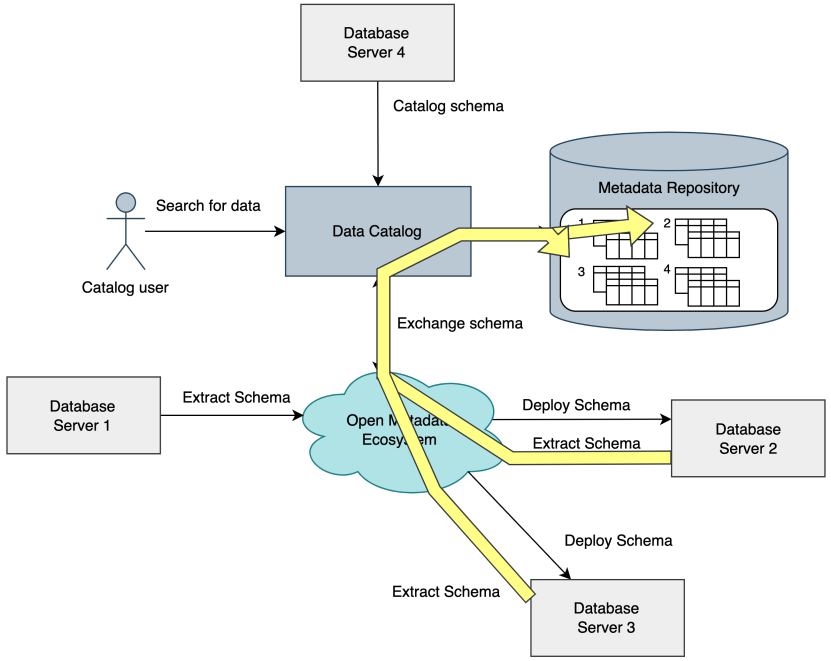
As the metadata capture in the open metadata ecosystem improves,
the data catalog directly benefits.



How should this metadata be updated?

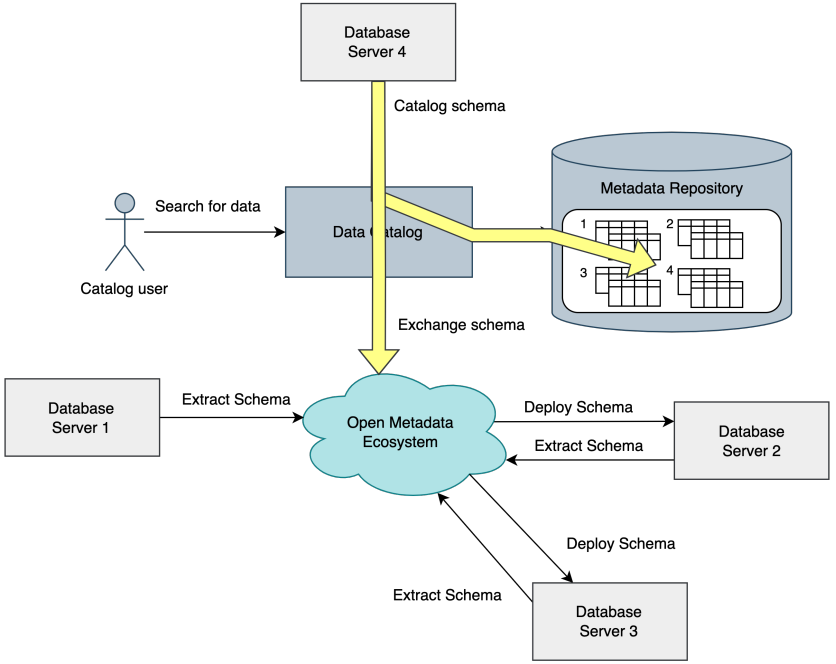
- Schema change in Database Server 1



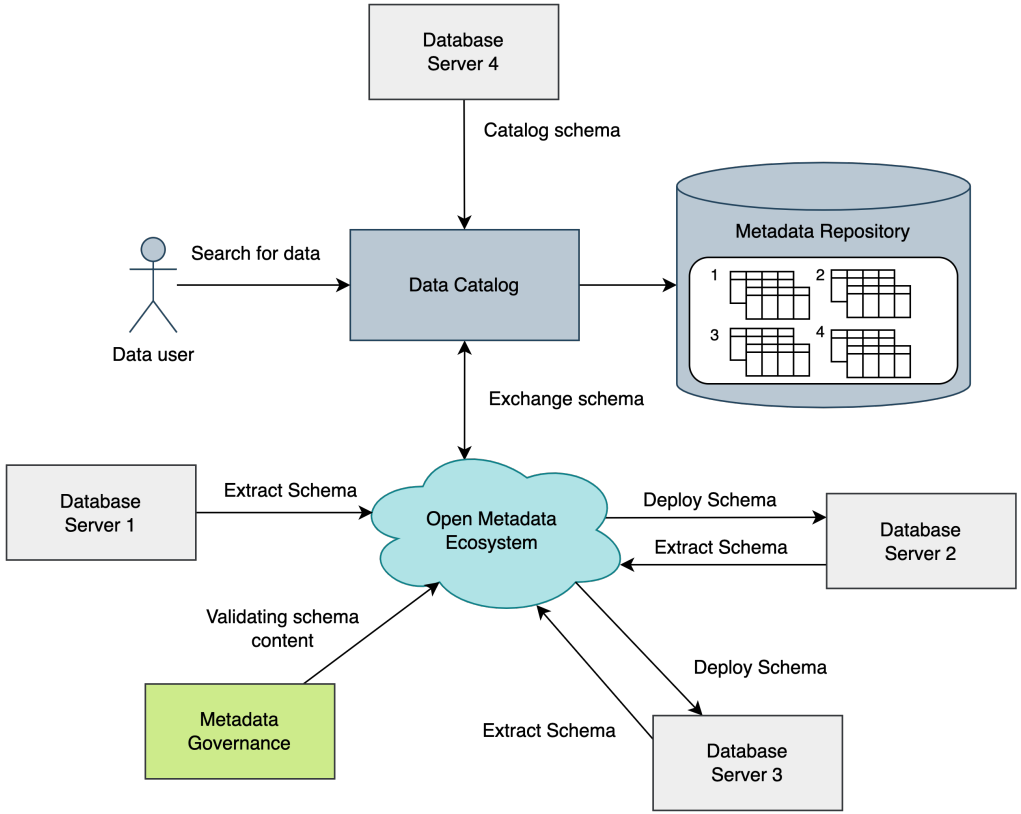


How should this metadata be updated?

- Schema change in Database Server 4

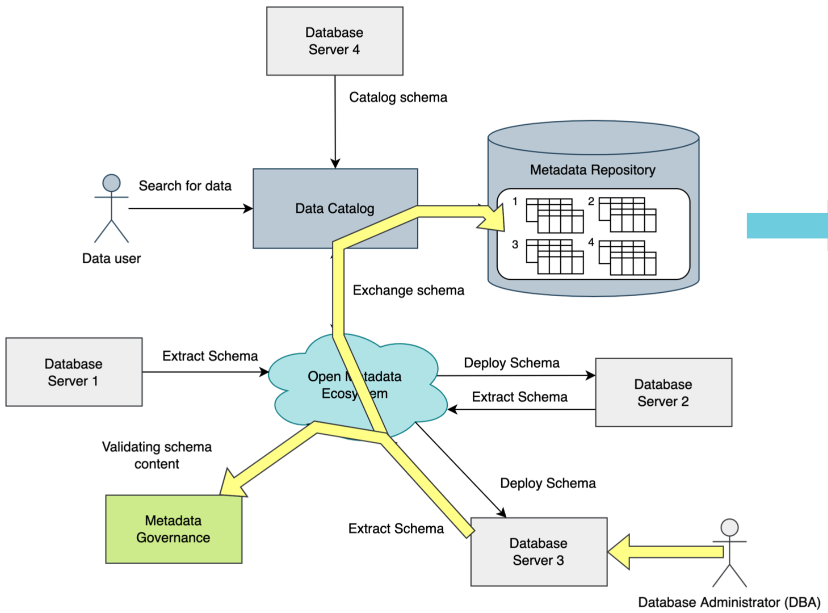


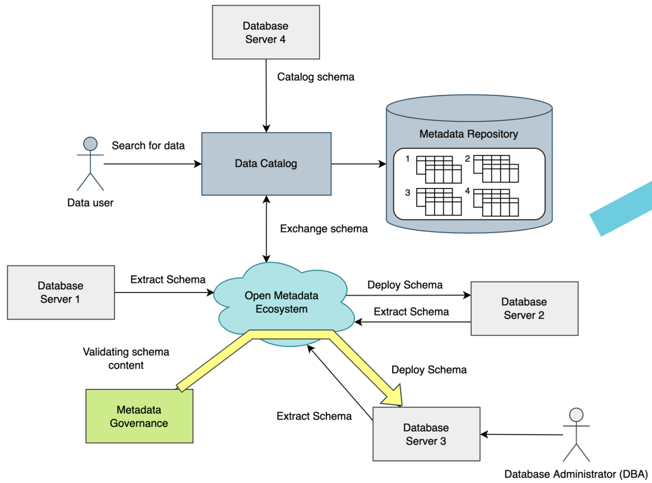
Metadata assurance also improves trust in metadata

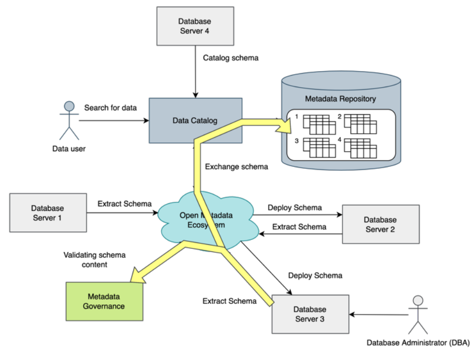


How should this metadata be updated?

- Schema change in Database Server 3 by DBA





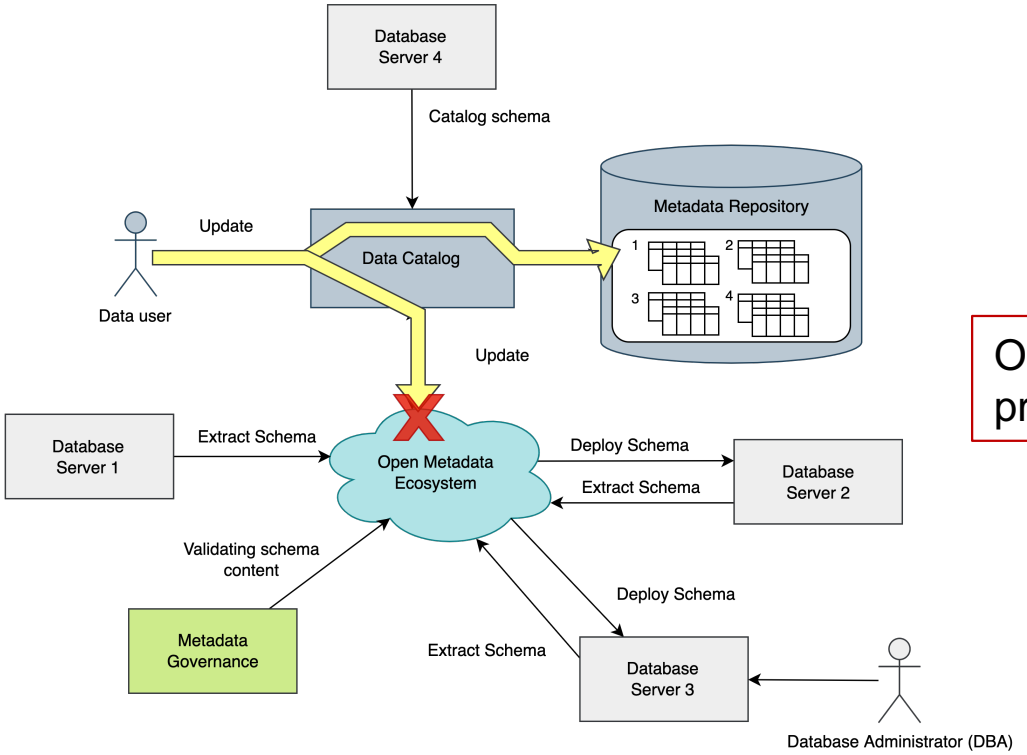


Metadata update triggers governance

EGERIA

How should this metadata be updated?

- Schema change in Database Server 1 by Data Catalog User

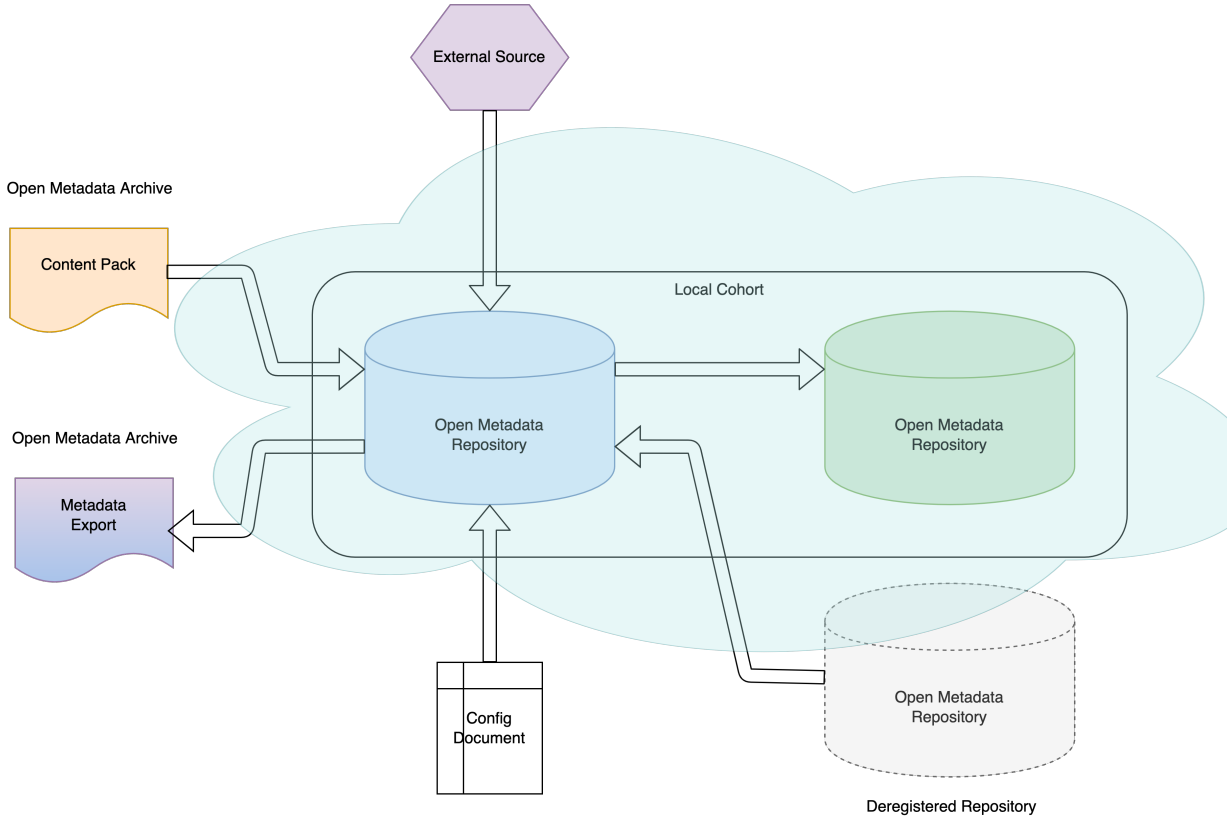


Open metadata provenance prevents update

EGERIA

Open Metadata Provenance

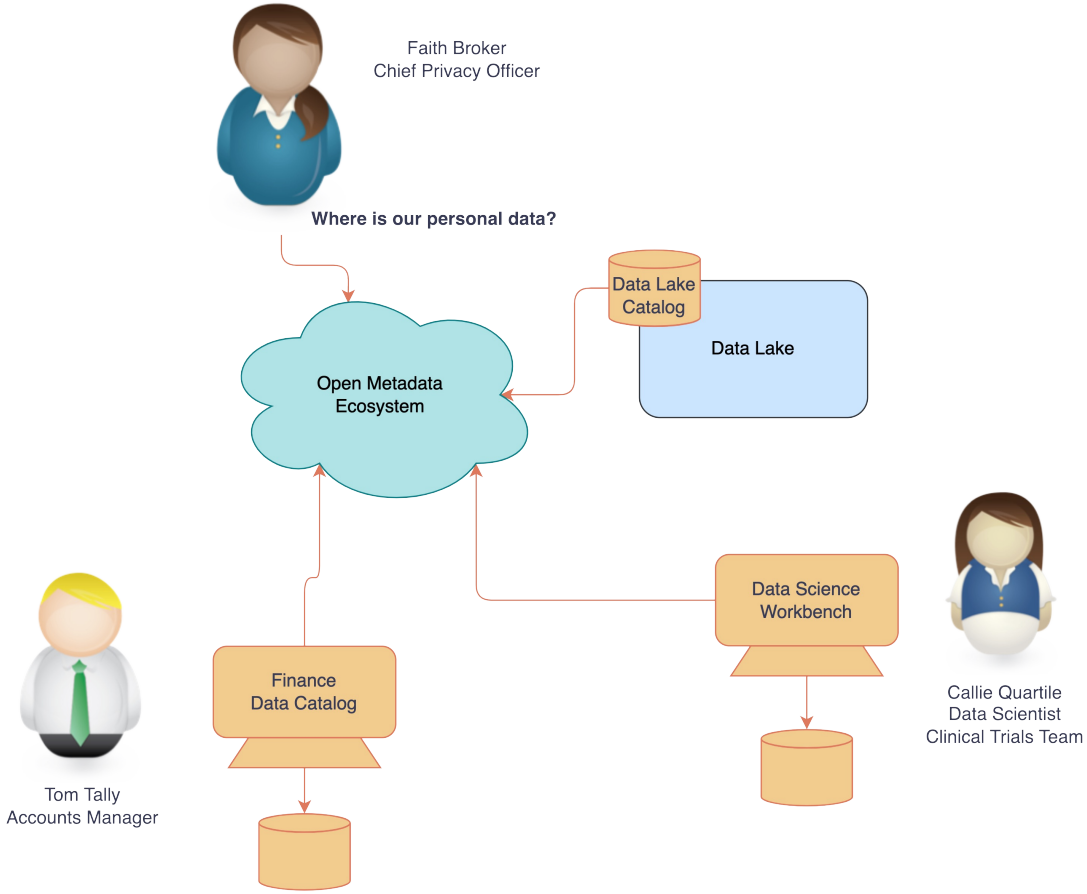
- The metadata collection where an element is created is its ‘home’
- Any copy of this element in another metadata collection is a read-only ‘reference-copy’



Coco Pharmaceuticals Use Case

Why integrate catalogs together?

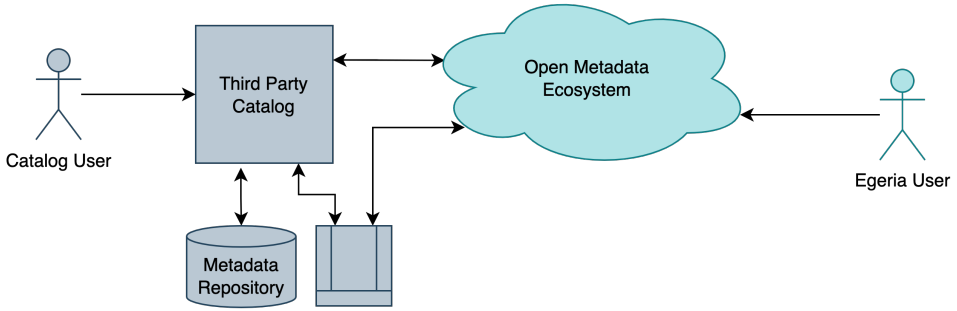
- Access to a broader collection of metadata from preferred tools



Connector comparison

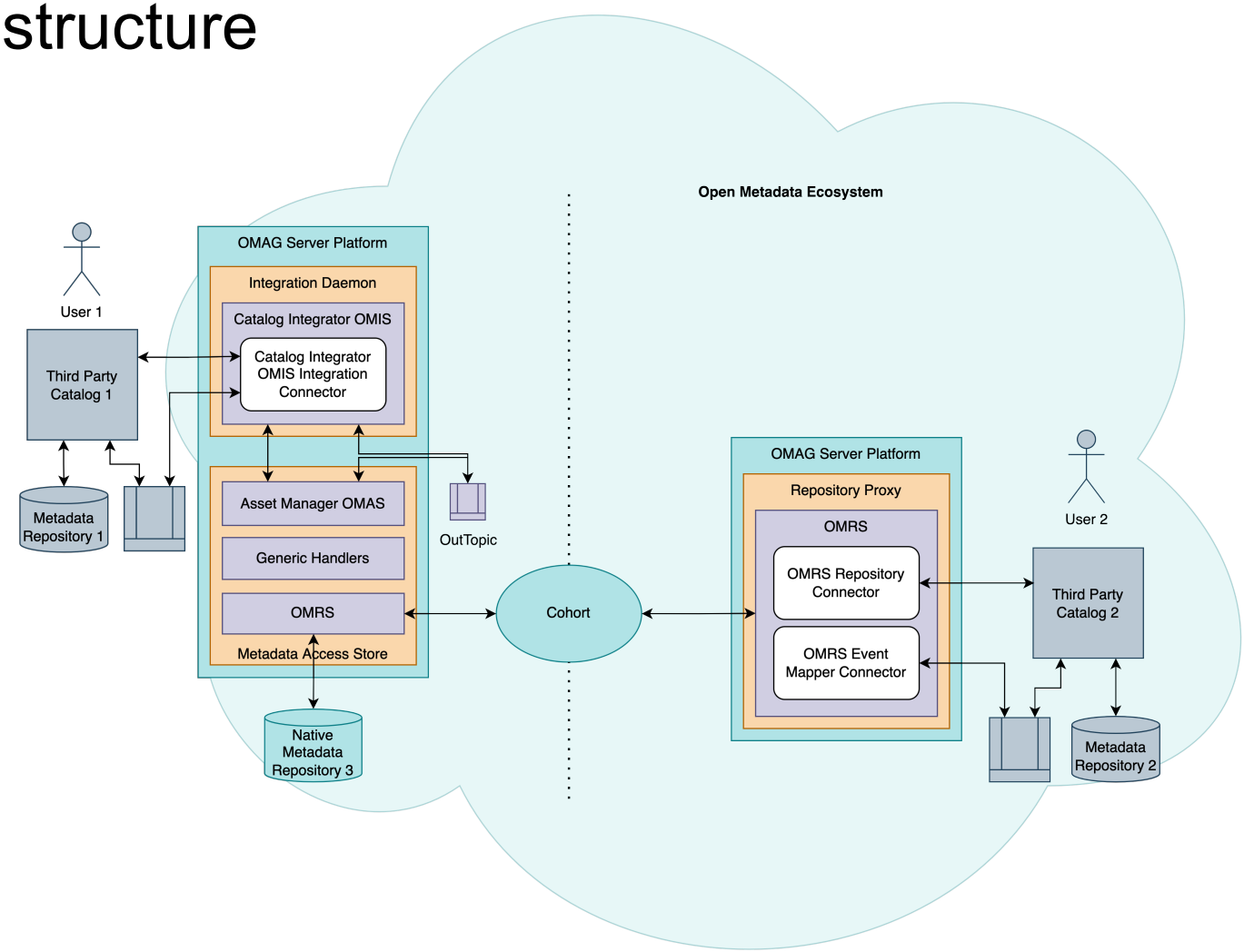
The challenge

- How should you connect a third-party data catalog to the open metadata ecosystem?
- Choices
  - Via an integration connector?
  - Via a repository connector?



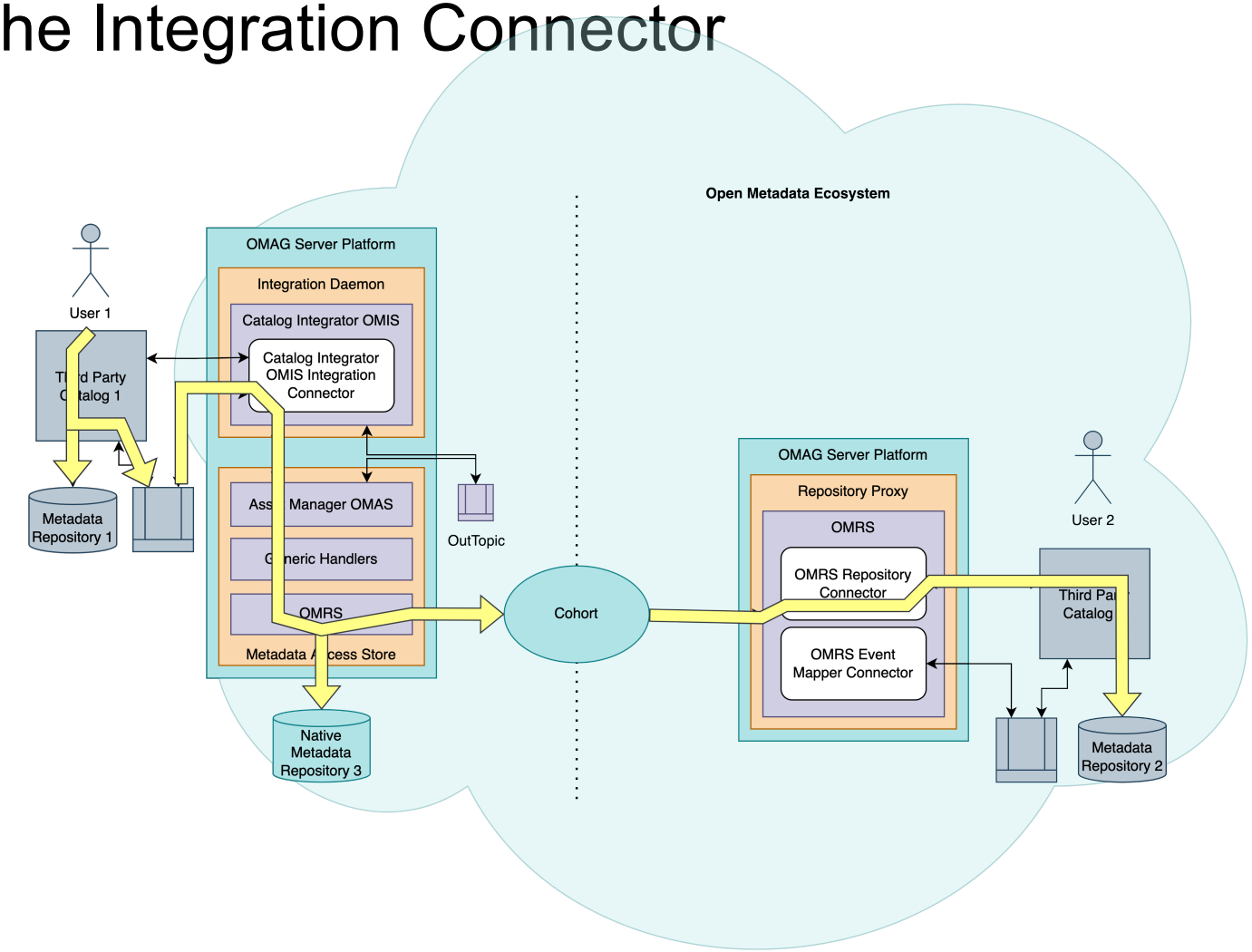
Comparison of infrastructure

- Integration connectors run in an integration daemon connected to a metadata access store
- Repository connectors run in a repository proxy directly connected to one or more cohorts.
- **User 1** works with metadata stored in **metadata repository 1**
- **User 2** works with metadata in **Metadata Repository 2**



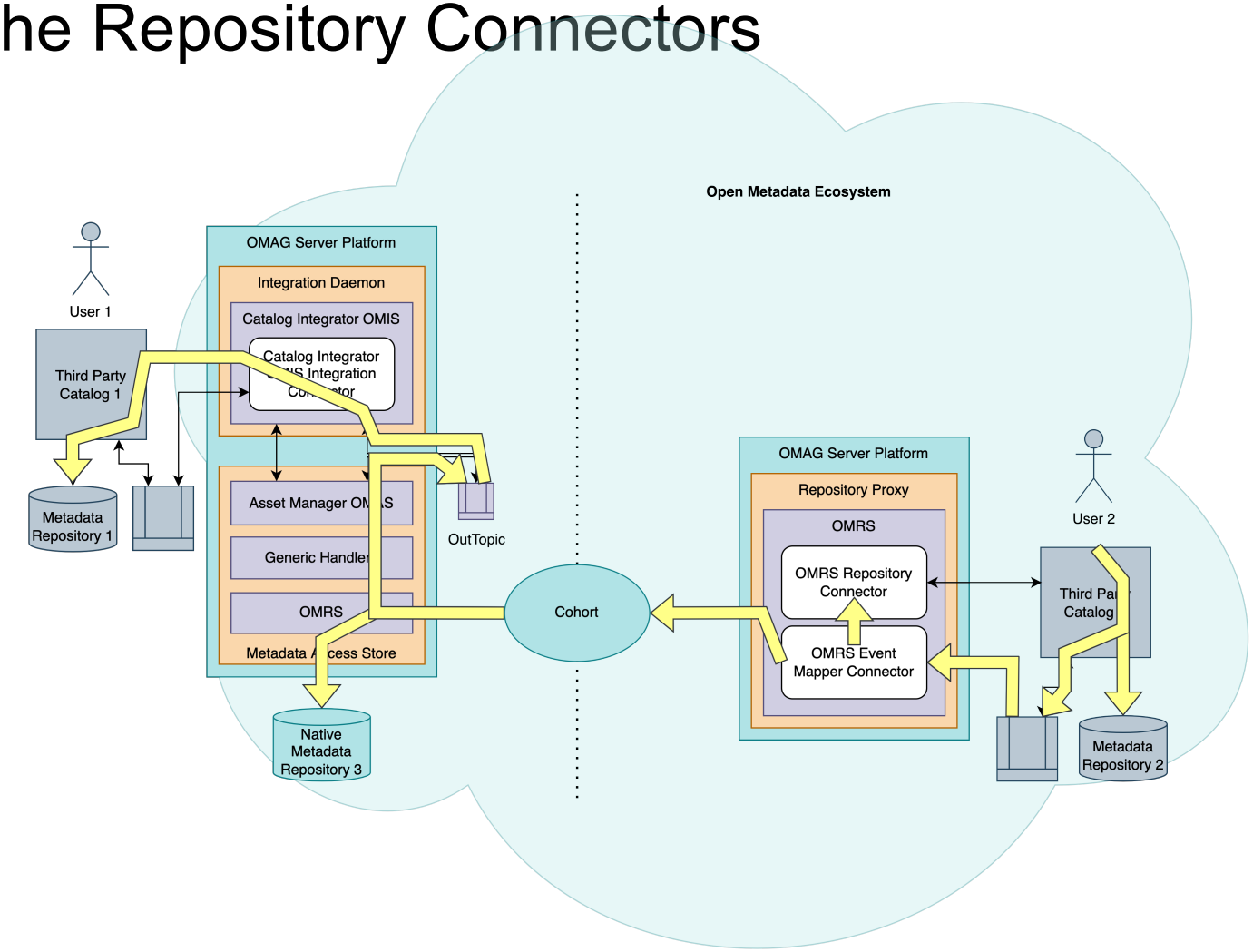
New metadata via the Integration Connector

- ***Native metadata repository 3*** maintains a copy of ***metadata repository 1***.
  - The integration connector chooses which of these repositories is the home directory
- Metadata copied into ***metadata repository 2*** is a reference-copy

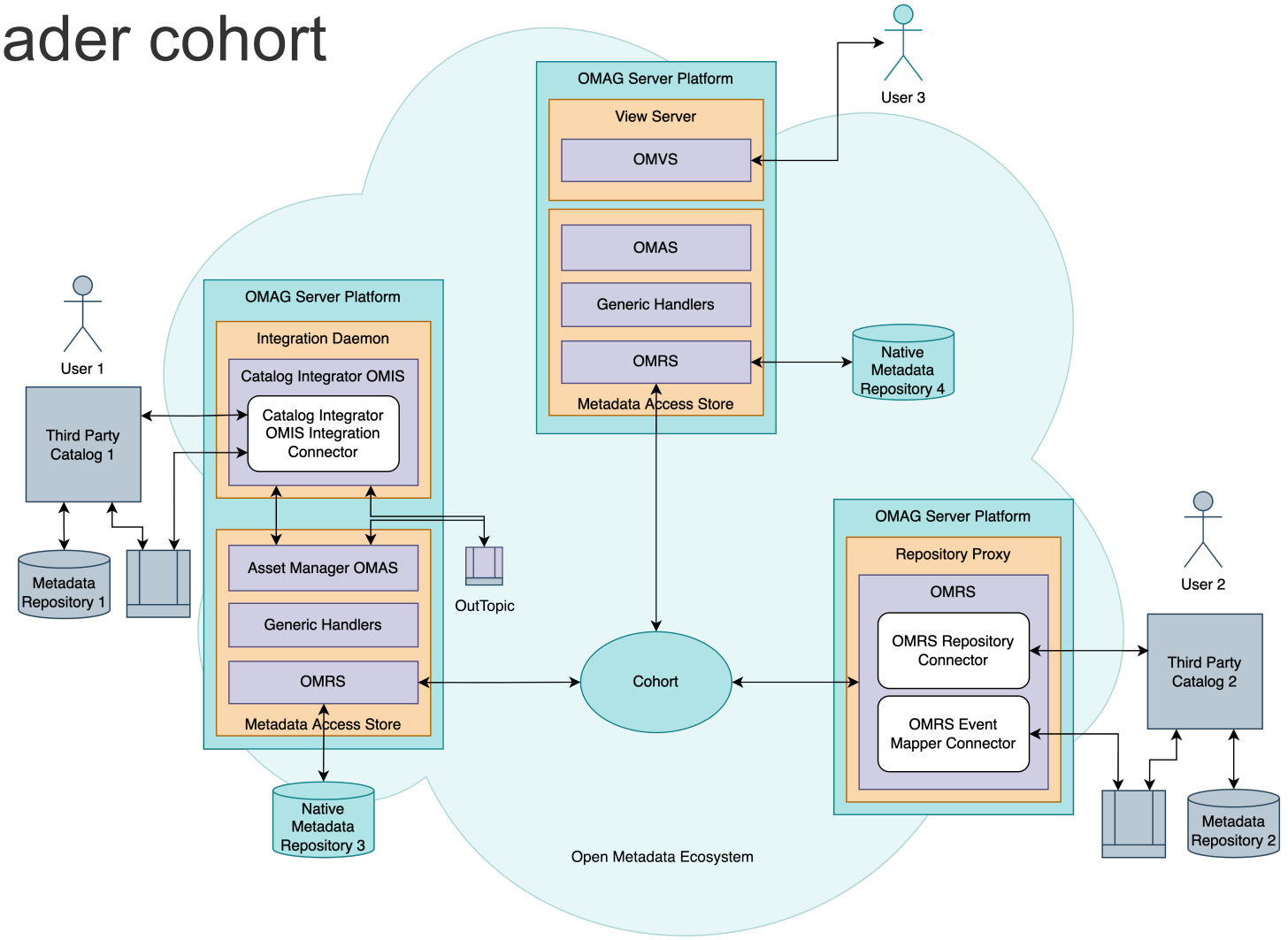


New metadata via the Repository Connectors

- ***Native metadata repository 3*** maintains a copy of ***metadata repository 1***.
  - The integration connector chooses which of these repositories is the home directory
- Metadata copied into ***metadata repository 2*** is a reference-copy

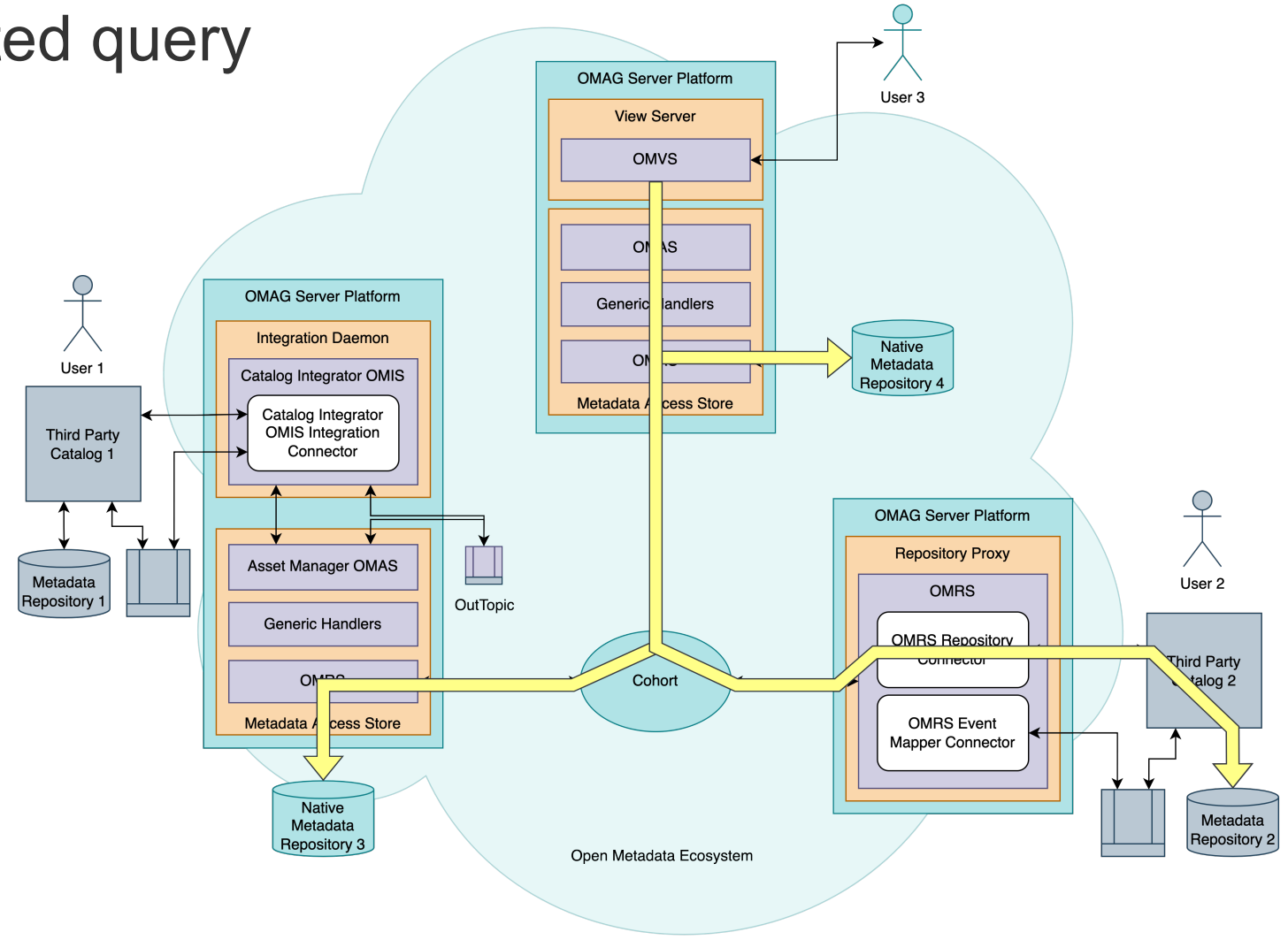


The broader cohort



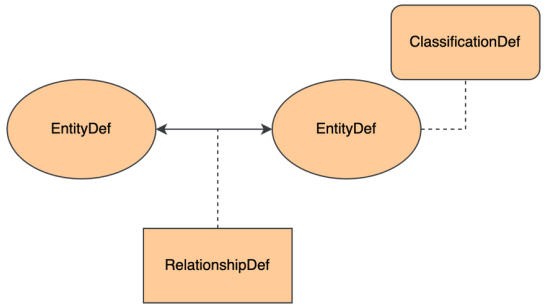
EGERIA

Federated query

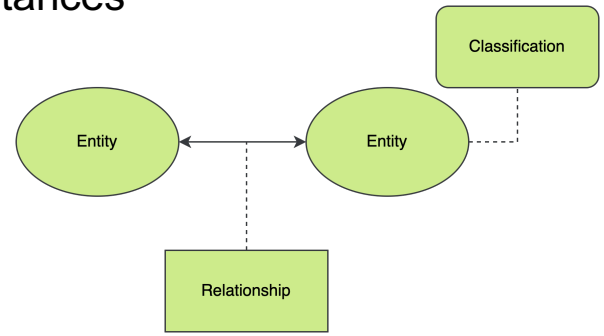


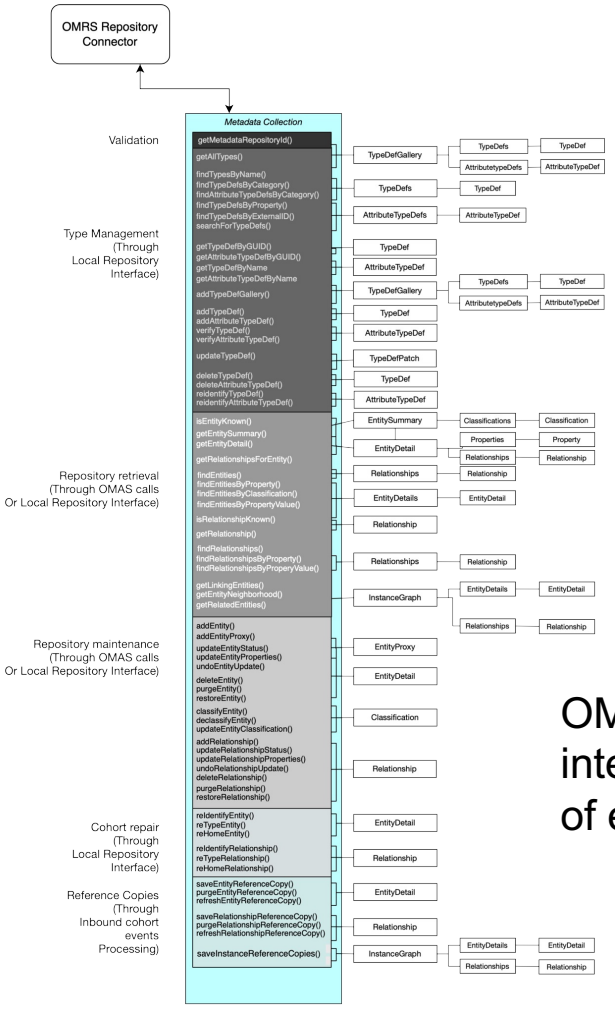
API comparison - OMRS

Types



Instances

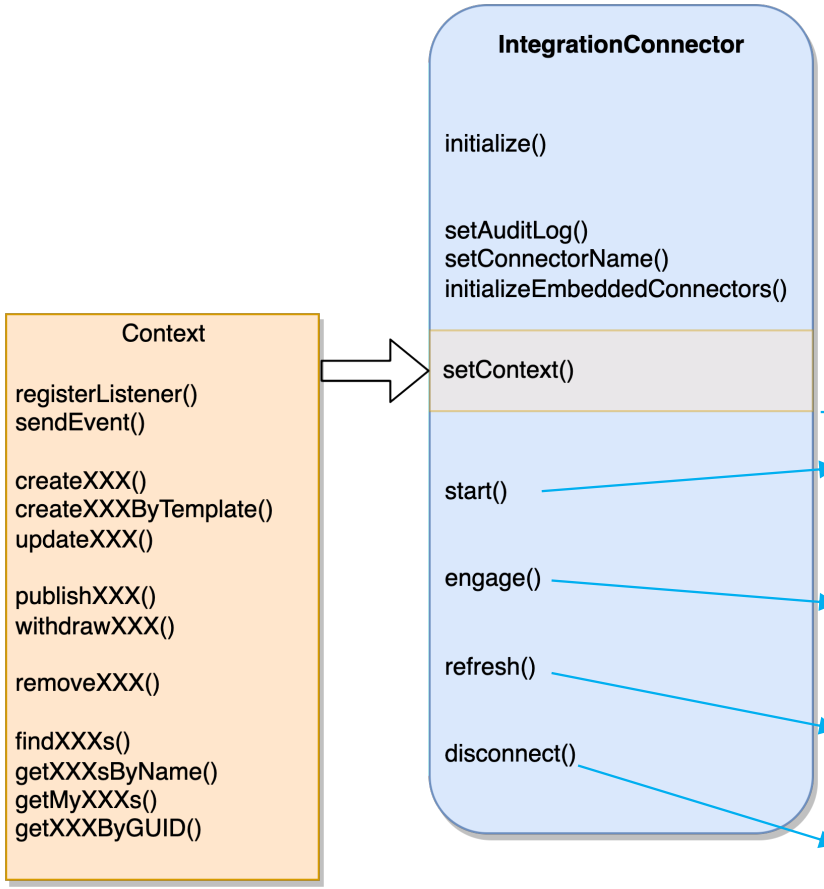




API is fine-grained repository API

OMRS handles metadata integrity and coordination of exchange

Integration Connector Implementation



For saving connector instance identifier and connection object.
Called from the ConnectorBroker.

Provides the logging destination.
Provides the name of the connector for logging.
Initializes embedded connectors

Handled by base class

Set up the integration service specific context.

Indicates that the connector is completely configured and can begin processing. This call can be used to register with non-blocking services. For example it can register a listener with the OMAS Out Topic with the context.

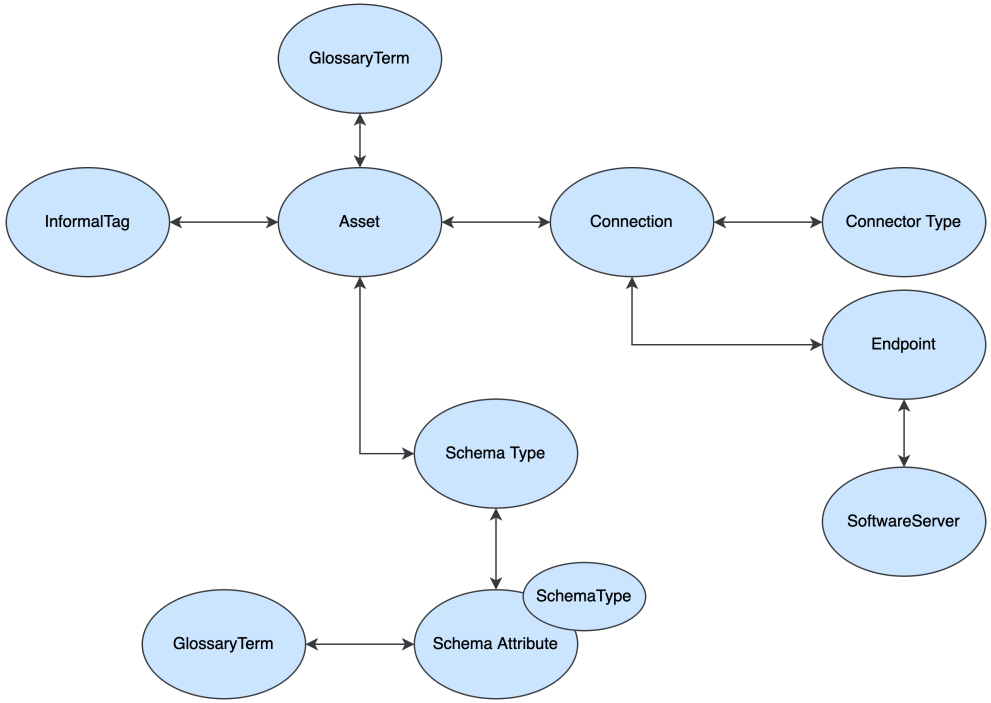
Used for blocking calls to wait for new metadata. It is called from its own thread iff the connector is configured to have its own thread. It is recommended that the engage() method returns when each blocking call completes. The integration daemon will pause a second and then call engage() again. This pattern enables the calling thread to detect the shutdown of the integration daemon server.

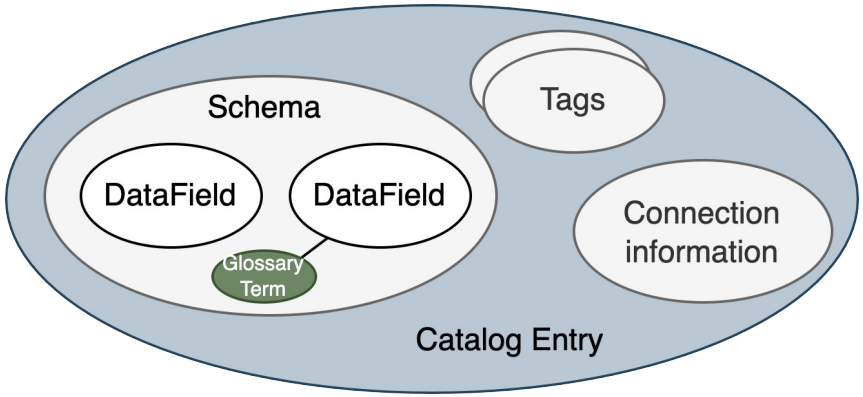
Requests that the connector does a comparison of the metadata in the third party technology and open metadata repositories. Refresh is called when the integration connector first starts and then at intervals defined in the connector's configuration as well as any external REST API calls to explicitly refresh the connector.

Free up any resources held since the connector is no longer needed.

API comparison - OMIS

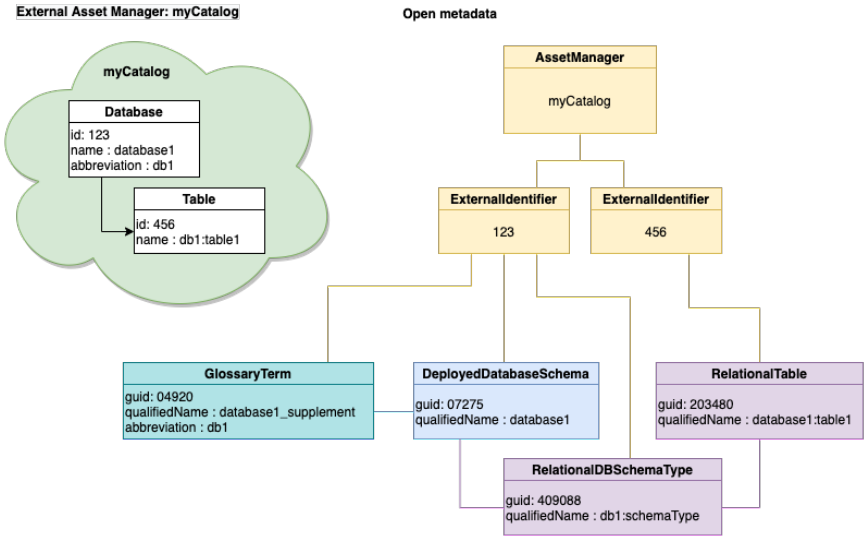
Catalog Integrator OMIS

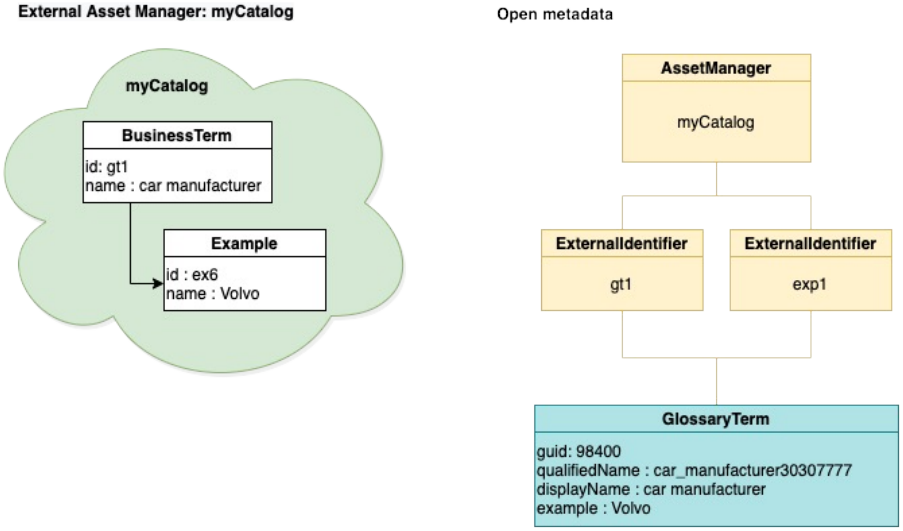




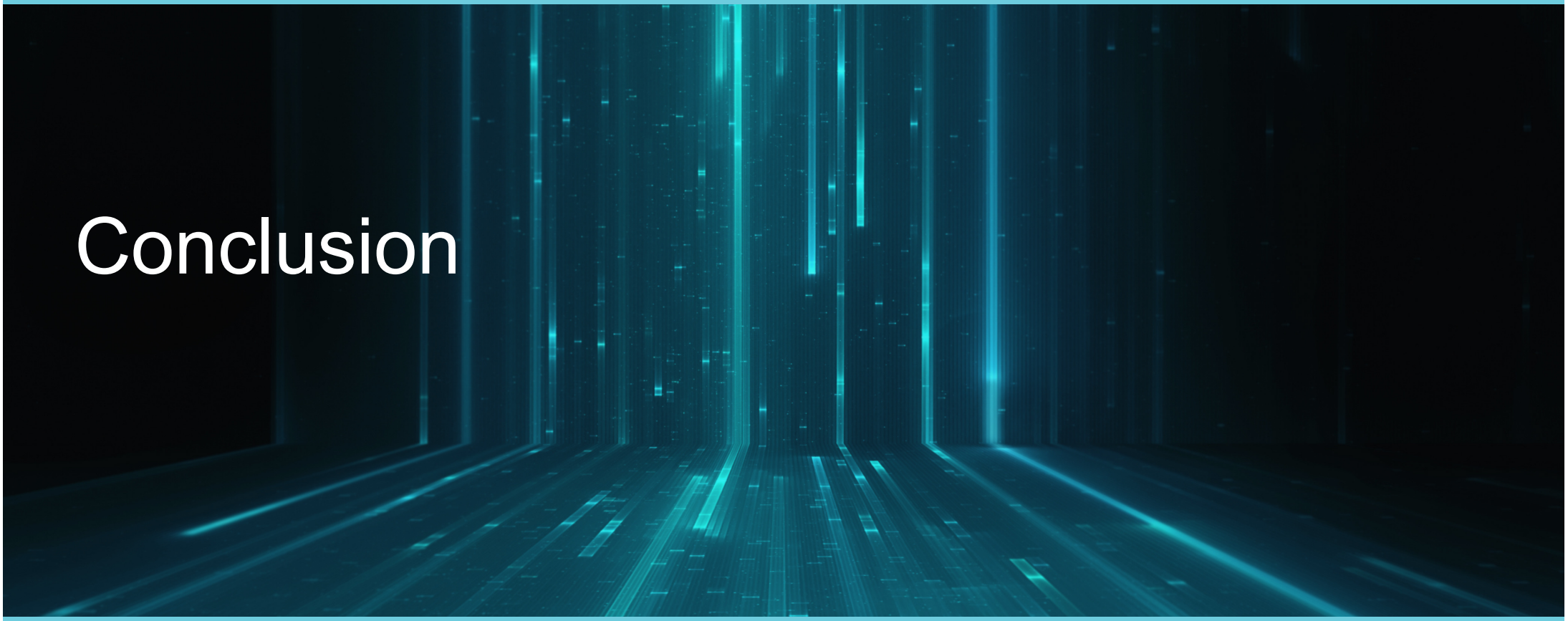
Example third-party data catalog structure

Using *External Identifiers* to manage complex mappings





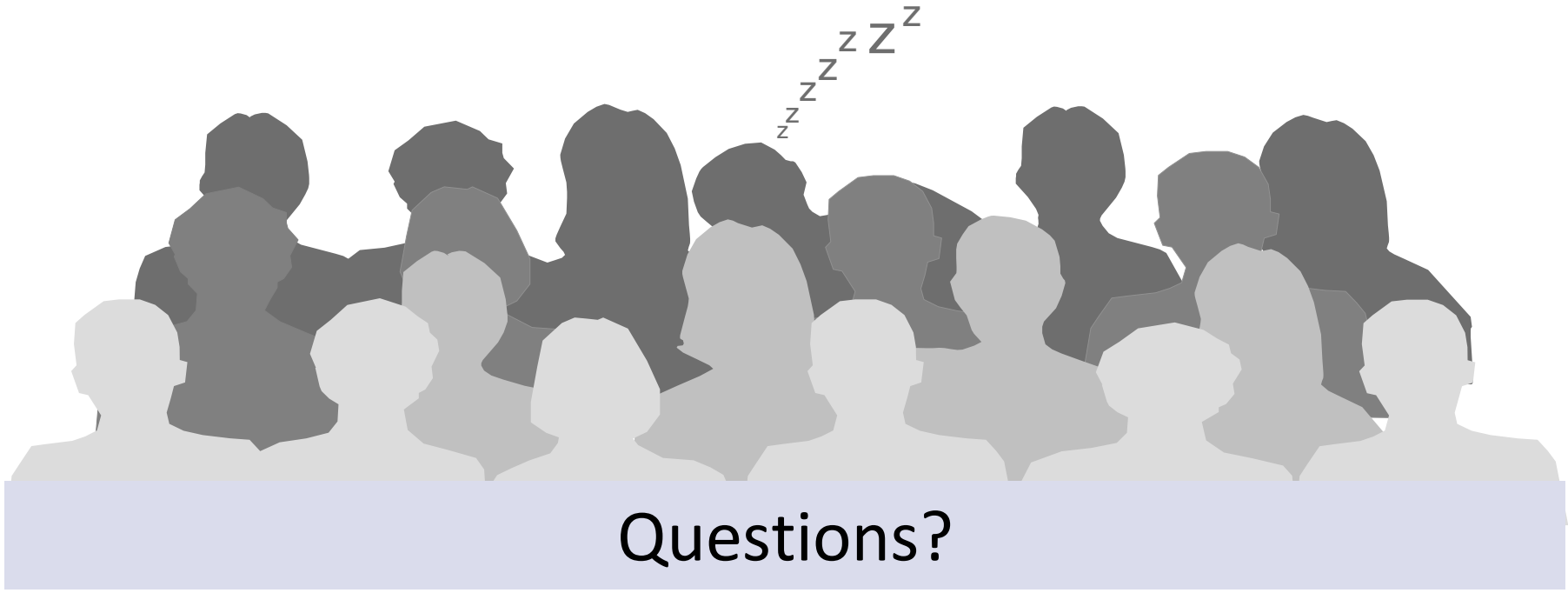
Conclusion



Conclusion

- Choosing the appropriate type of connector for your data catalog depends on the capability of the catalog and its intended usage
- Simple choices in favor of an integration connector
  - Will not/can not support federated queries due to API or capacity
  - Wildly different granularity of API from the OMRS
- Simple choices in favor of the repository connectors
  - Volume and rate of change of metadata makes a copy impractical
  - Sensitivity of metadata makes owners unwilling to share with no-one but a few trusted users
- Other considerations
  - Control of which metadata is shared
  - Control of update rights
  - Storing reference copies

Open forum



Egeria's webinar series

~~7th~~ **15th March 2022**	~~15:00~~ 14:00 UTC	**How to build an integration connector**	This session covers how to extend Egeria's automated cataloguing to include metadata from a new technology. It describes how automated cataloguing works and the role of the integration connector. It covers the design of the integration connector using examples to illustrate the different approaches and their benefits and and challenges. It shows how to set up a project for a new connector, how to build and package it and finally it shows the new connector running in Egeria. Zoom Conference https://zoom.us/j/523629111	**Mandy Chessell**
4th April 2022	15:00 UTC	**How to choose: Integration or repository connector?**	This session compares using Integration Connectors with Repository Connectors to connect technologies into Egeria. We will go through the pros and cons of integration connectors and both types of repository connectors (native and proxy) and how these choices impact and benefit your Egeria eco-system. Zoom Conference https://zoom.us/j/523629111	**Mandy Chessell**
9th May 2022	15:00 UTC	**Using a repository connector**	This session covers how to use Repository Connectors to connect technologies into Egeria; focussing on XTDB (formerly known as crux). Ever wanted to know what the state of your metadata was at some specific time in the past? This session will introduce the XTDB open metadata repository that supports these historical metadata queries. Zoom Conference https://zoom.us/j/523629111	**Chris Grote**
6th June 2022	15:00 UTC	**Kubenetes operators and Egeria**	This session will cover how easy it is to run Egeria in Kubenetes and how the Egeria Kubenetes operator can be used to manage Egeria in a Kubenetes environment. Zoom Conference https://zoom.us/j/523629111	**Nigel Jones**

THANK YOU!

EGERIA

Achievements

- 700 linked open metadata types demonstrating how the knowledge from many tools can be linked together.
- Open metadata repository interface proven for table, graph and hierarchical DB stores.
- Enterprise queries and replication across heterogeneous technologies
- Conformance test suite and mark
- Automated configuration of data virtualization technology and security as new data sets are added to a data lake
- Suite of persona-based labs and tutorial using Jupyter Notebooks.
- Virtual graph of metadata maintained across distributed heterogenous metadata repositories.
- Frameworks, APIs and connectors for minimizing integration cost for different types of technologies
- Virtual repository explorer UI
- Instance based security
- Controlling visibility of assets through zones
- Scalable, secure platform configurable and customizable through connectors
- Purpose-based data access
- Metadata versioning and provenance
- Multi-tenant UI based on carbon
- W3C semantic standards pattern for data model exchange
- Automation of metadata acquisition through templates, daemons, discovery services and stewardship.
- Classification of assets
- Reference data management
- Multi-technology collaboration and feedback
- Multi-domain governance model
- Digital service lifecycle, from business design, development, devOps and use.
- Comprehensive open lineage services.
- Metadata deduplication