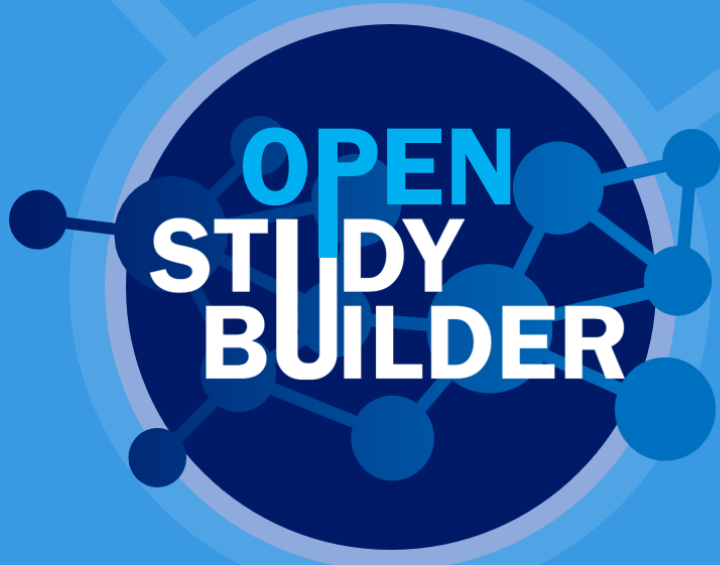


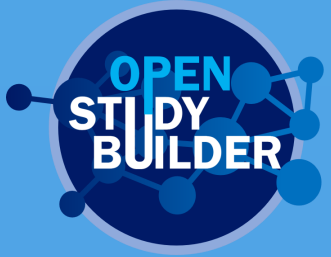
OpenStudyBuilder Hub (OSB-Hub)

OSB-Trail-SystemEngineering

Innovating Through Community Collaboration
April, 28th 2025

Pascal Bouquet





Agenda

Introduction

- Introduction - Pascal Bouquet
- Data Loading - Marius Conjeaud
- Data Migration - Marius Conjeaud
- Helm Chart for OSB Deployment— Gerard Castillo

Purpose of OSB Hub



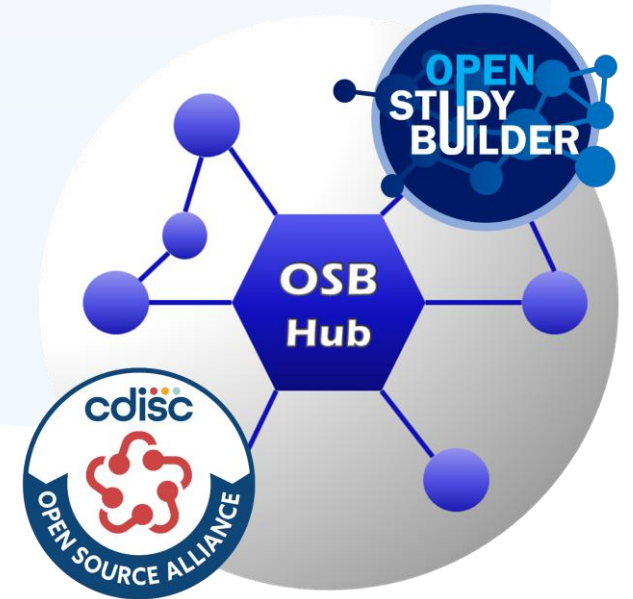
COSA Community

- List, add and discuss use-cases
- Feedback & community interest
- Prioritise use-cases of interest
- Manage and run focus projects
- Utilization and enhancement of the OpenStudyBuilder

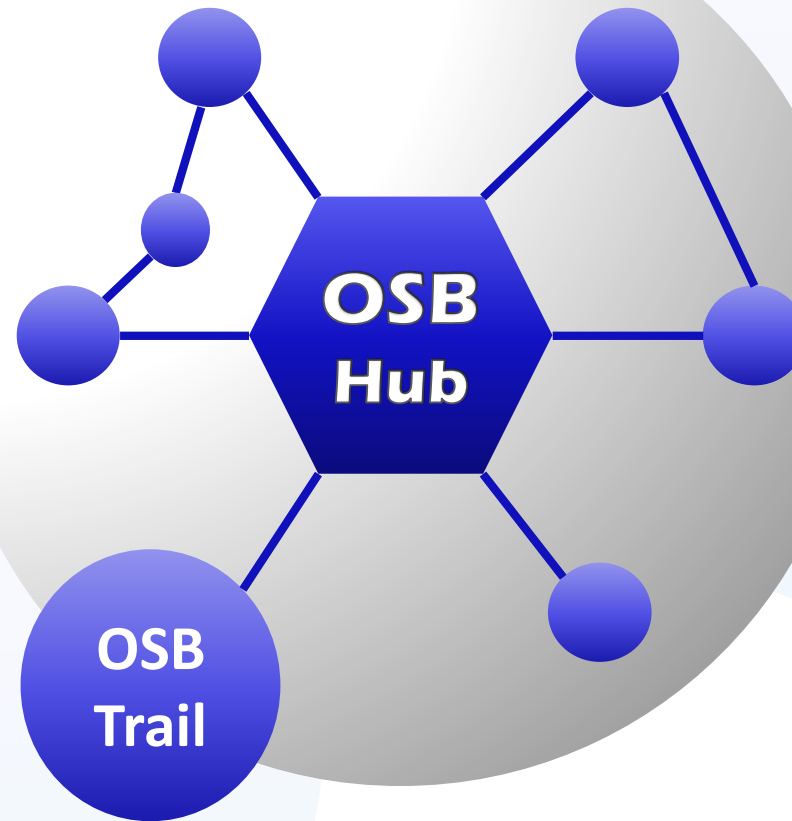
😊 Join us on Slack: [Invite](#)

💡 Feedback on Use-Cases: [Discussions](#)

📄 Checkout Information: [Wiki](#)



OSB Trails – Focus Projects



OSB-Trail-SystemEngineering: Objectives



OSB-Trail-System Engineering: SCOPE

Priority1 Scope

1. Optimizing deployment workflows for OSB across diverse environments. First deployment and deployment of new release
 - ✓ Cloud: Azure, AWS, Google
 - ✓ Containerization of OSB: EKS, AKS, GKE, OpenShift, Vmware Tanzu, Fargate, ACI ...
 - ✓ Develop a Terraform-based deployment template for easy replication across cloud environments (AWS, Azure, GCP).
2. Exploring and implementing robust authentication methods tailored to OSB.
3. Facilitating seamless integrations between OSB and other enterprise systems.
4. Inventory of OSB implementations with their technical deployment pattern
5. Monitoring of OSB/Observability
6. Archiving, Back-up, DR
7. Share Performances/Scalability Testing
8. Validation approach of OSB and Interest in developing UI testing tools to support validation.
9. Applying GraphRag on OSB
10. Changing OSB CSS



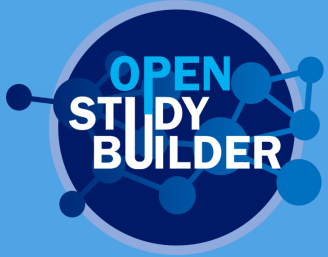
Points discussed in March 24th Meeting

(1/2)

- API sits between the Neo4j graph database and the front end.
- RESTful, JSON-based APIs callable from Python, R, SAS, Word, etc.
- Supported via Swagger interface for testing and documentation.
- Use Cases presented by Carlos:
 - Examples shown for accessing library data (e.g., control terminology) and study data (e.g., design, SOA).
 - Demonstrated how to use APIs for version control, audit trails, and data export.
 - Emphasis on standards control and auditability within study design.
- Authorization and Roles:
 - Explained OSB user roles (study/library reader/writer, admin).
 - API access is permission-based and requires authentication.

Points discussed in March 24th Meeting

- Consumer API:
 - A new, simplified API layer meant for external tools (e.g., Word add-ins, EDC integration).
 - Current features are limited but expanding.
 - Preferred API for third-party integrations due to future stability and versioning.
 - Limitations and Improvements
 - Known performance issues with some endpoints.
 - Planned API versioning for consumer API.
 - Consumer API will help decouple front-end-specific needs from external user needs.
-



Agenda

Introduction

- Introduction - Pascal Bouquet
- Data Loading - Marius Conjeaud
- Data Migration - Marius Conjeaud
- Helm Chart for OSB Deployment— Gerard Castillo

Introduction

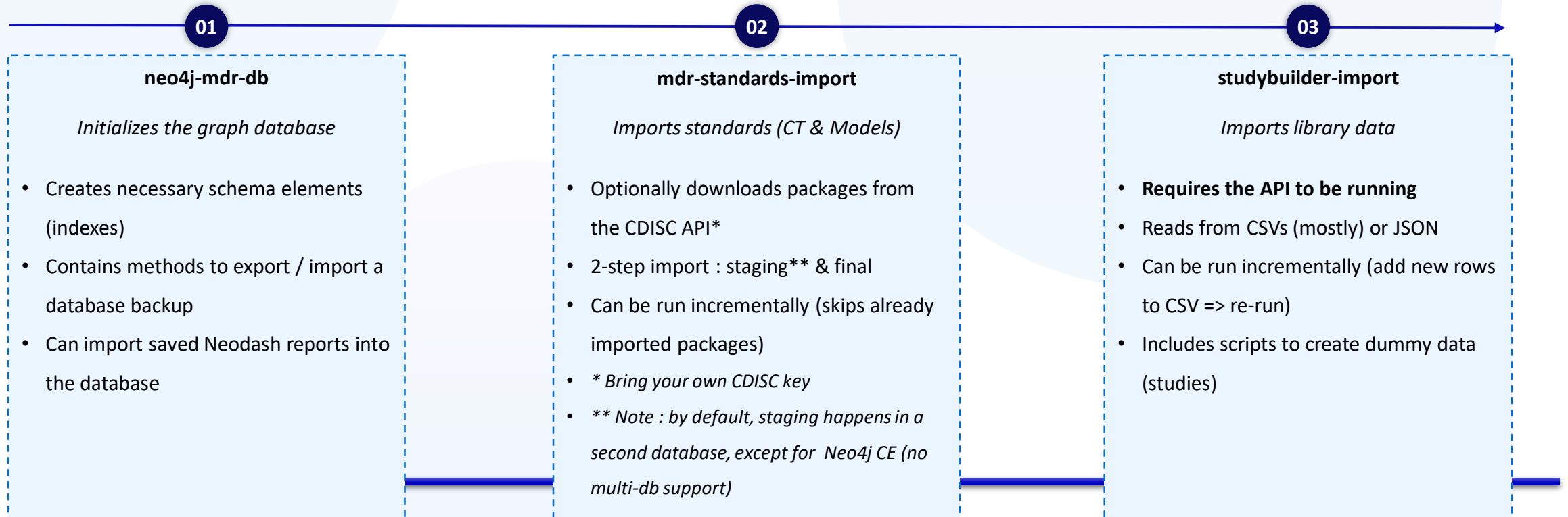
OpenStudyBuilder stores data from various sources, with different lifecycles

- 01 Standards data (CT, Data Models & IGs, Dictionaries,...)
- 02 Sponsor library data (Activity Concepts, Units, Sponsor CT & Models,...)
- 03 Optionally, legacy protocols
- 04 "Live" inputs (OSB-built protocols, new templates,... Everything added after the initial import)

Initial data loading

OSB provides a sequence of steps set-up your environment with data

The various steps are stored in different repositories



Deep dive - live

Problem

OpenStudyBuilder stores data from various sources, with different lifecycles

- 01 Standards data (CT, Data Models & IGs, Dictionaries,...)
- 02 Sponsor library data (Activity Concepts, Units, Sponsor CT & Models,...)
- 03 Optionally, legacy protocols
- 04 "Live" inputs (OSB-built protocols, new templates,... Everything added after the initial import)

This can be scripted

What about this ?

Initial data can always be re-imported / incrementally imported ; but data created by users of the application must be preserved as-is.

=> What happens when a new release comes out ?

Apply schema changes and data corrections to production data

Migrations can convert existing data from one "version" of the data model to another

4. User Identification

Change Description

- Use `author_id` instead of `user_initials` field on all relevant nodes/relations to identify users performing actions.
- Retrieve IDs of all relevant users from Active Directory based on their initials.
- Create `User` nodes for all unique users that have performed any actions until now, and persist their `id`, `email/username`.
- [Related PR.](#)

Nodes and Relationships Affected

- `CTPackage`, `StudyAction`, `Edit`, `Create`, `Delete` nodes: `user_initials` field value copied to `author_id` field.
- `HAS_VERSION`, `HAS_TERM`, `HAD_TERM`, `LATEST_DRAFT`, `LATEST_LOCKED`, `LATEST_RELEASED` relations: `user_initials` field value copied to `author_id` field.
- `User` : new nodes created

5. Unifying StudyVisit window units for all StudyVisits in a Study

Change Description

- All StudyVisits should have the same window unit across all StudyVisits in a single Study.
- If some Study used `week` unit for some of the StudyVisits it should be changed for `days`
- [Related PR.](#)

Nodes Affected

- `:StudySelectionMetadata`

How does it work ?

The db-schema-migration repo provides:

- ⑩ Migration scripts, with overview files to explain what they are doing `/migrations`
- ⑩ Data correction scripts, with overview files `/data_corrections`
- ⑩ Tests to verify that the migrations are working correctly and do not introduce unwanted side effects. `/tests`
Several notes here:
 - Any migration published to OSB has been validated on Novo Nordisk's production data
 - Tests require test data. The ones used for testing before releasing to OSB is provided in `/tests/data`
 - You can create your own test data (see the repo README for instructions) if you want to verify for your context
- Some operations (only the data corrections so far) are wrapped in a `@capture_changes` decorator. This is a Neo4j Enterprise-only feature and logs all changes made to the data, so that you can validate what happened during the correction.
- *Note: db-schema-migration requires some repo to be added as submodule - this is not possible today in the current setup. You either need to set this up yourself or "inline" the repos.*

Deep dive - live

Limitations

Migration status tracking

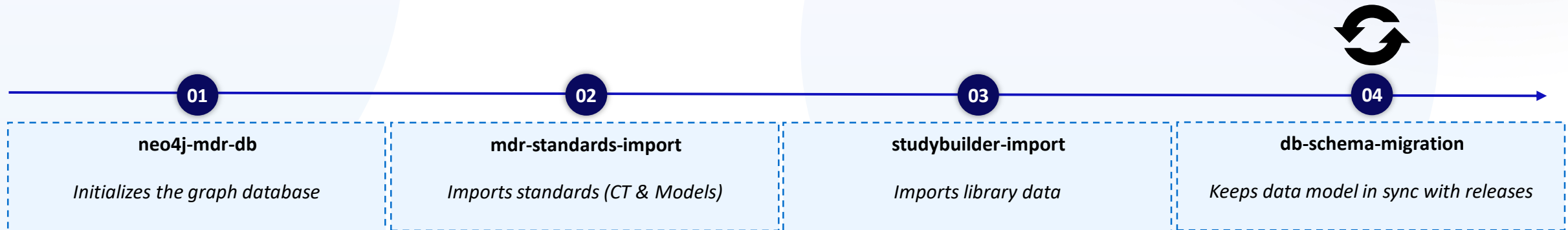
The current Migration process does not provide a "data model version" tag. This means you have to come with your own way of tracking which migration was last ran on your database.

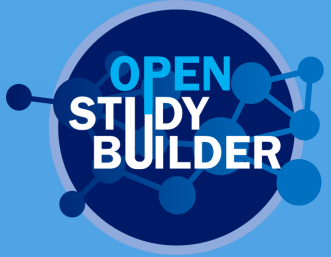
Note: the migration files do include some information about which version they were meant for, but it shows the internal NovoNordisk StudyBuilder tag

```
"" Schema migrations needed for release 1.11.0 to PROD post December 2024.""
```

Conclusion

Recap





Agenda

Introduction

- Introduction - Pascal Bouquet
- Data Loading - Marius Conjeaud
- Data Migration - Marius Conjeaud
- Helm Chart for OSB Deployment— Gerard Castillo



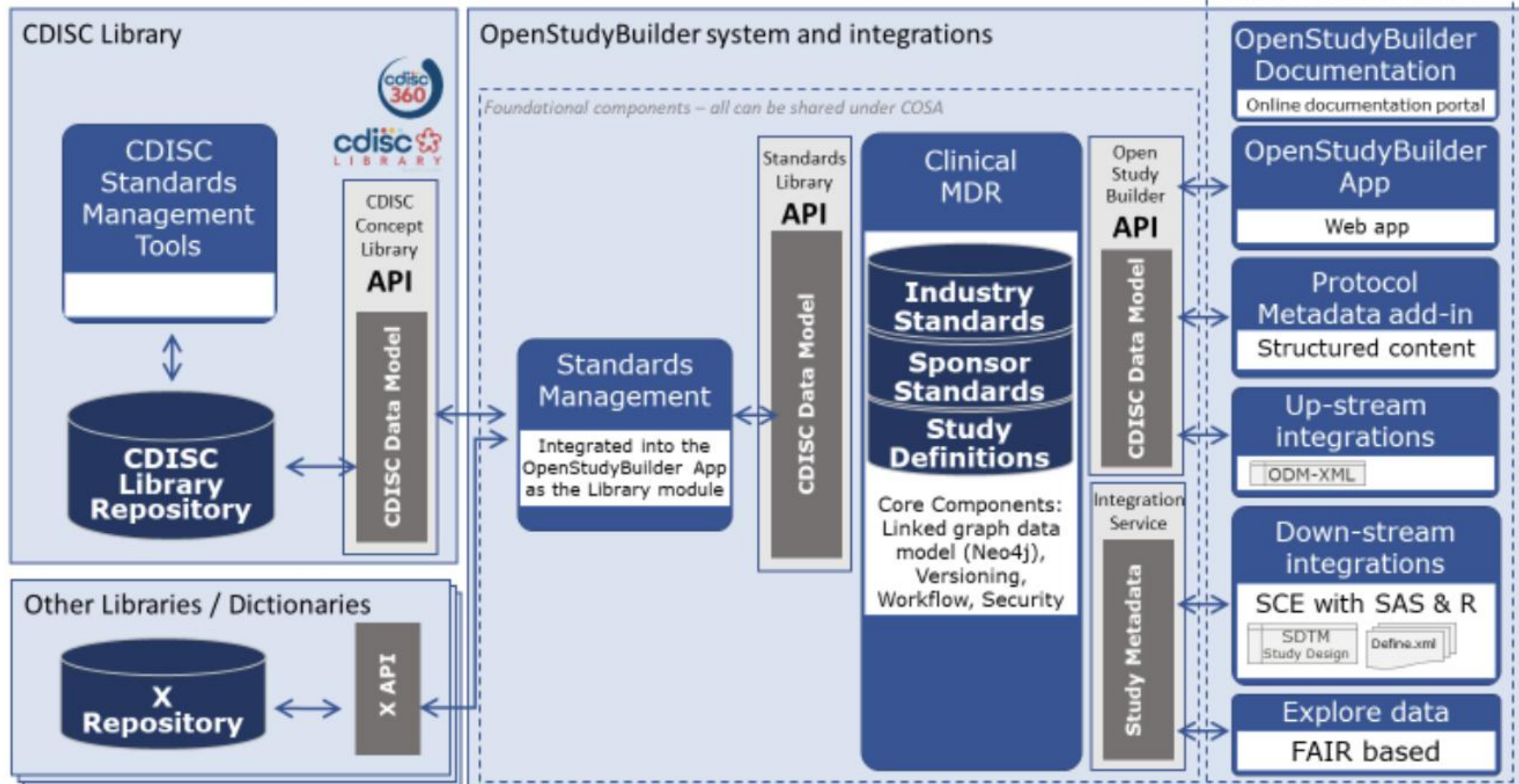
Next Meeting

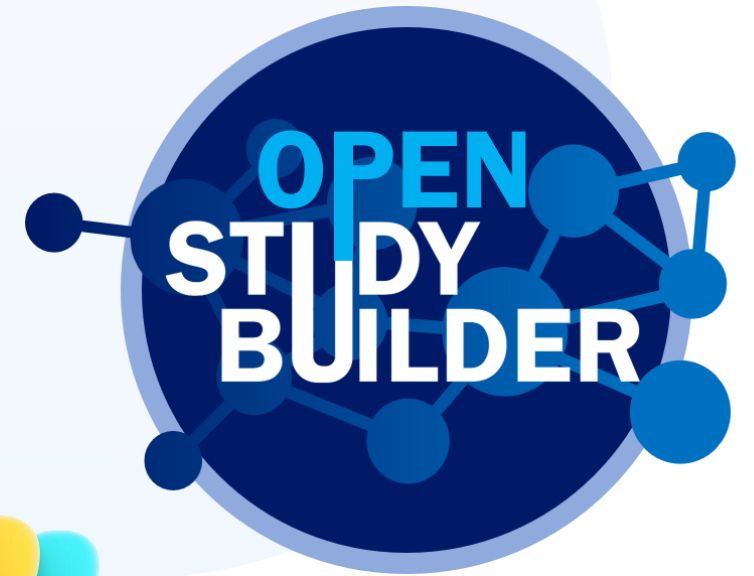
June 2nd 2025 - 4 pm CET

Potential topics:

- 1. Testing approach**
- 2. USDM loading in OSB**

OpenStudyBuilder Conceptual Architecture





**We need YOUR
Feedback!**



OSB-Trail-System Engineering: References

1. [OpenStudyBuilder Gitlab](#)
2. OpenStudyBuilder Installation
 - ✓ Comprehensive information on various aspects of the application, including installation procedures: [OpenStudyBuilder Documentation Portal](#)
 - ✓ [Developer Setup Guide](#)
3. We can integrate with OSB through its REST APIs. Here are some useful links:
 - ✓ OpenStudyBuilder APIs: [API - OpenStudyBuilder](#)
 - ✓ OpenStudyBuilder Authorization setup: [clinical-mdr-api/doc/Auth.md · main · Novo Nordisk / NN-Public / OpenStudyBuilder / OpenStudyBuilder-Solution · GitLab](#) It explains the code flow and shows that integration are implemented in the API. You basically need to request a token from your authority (e.g. OKTA), and then pass it along with your API calls for authorization.

COSA Collaboration Team



OSB Hub



Mission and Vision:

Support the utilization and enhancement of the OpenStudyBuilder open-source tool.



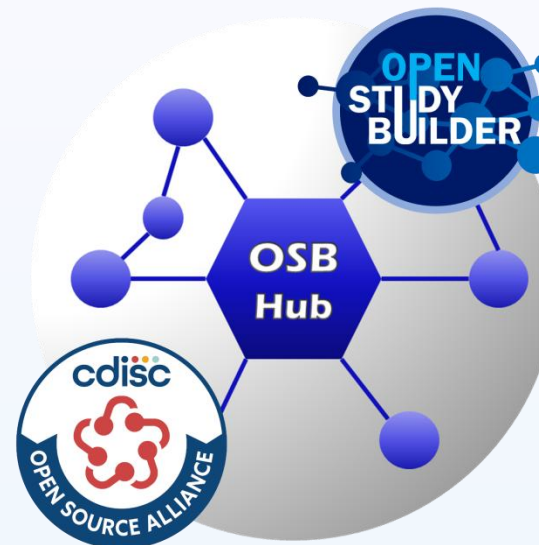
Core Objectives:

Collect feedback, run focused projects, drive innovation through community engagement.



Participants:

Everybody is invited!



OSB Trails – Focus Projects

Work as a community on:

- Enhanced documentation
- Best practices
- Additional requirements
- Supporting tools
- Concepts for integrations
- ...

Community powered Use Cases:

- Objective is to create a focused group for addressing technical challenges in deploying and operating OSB.
- Together, we'll chart the course for our upcoming trails.

