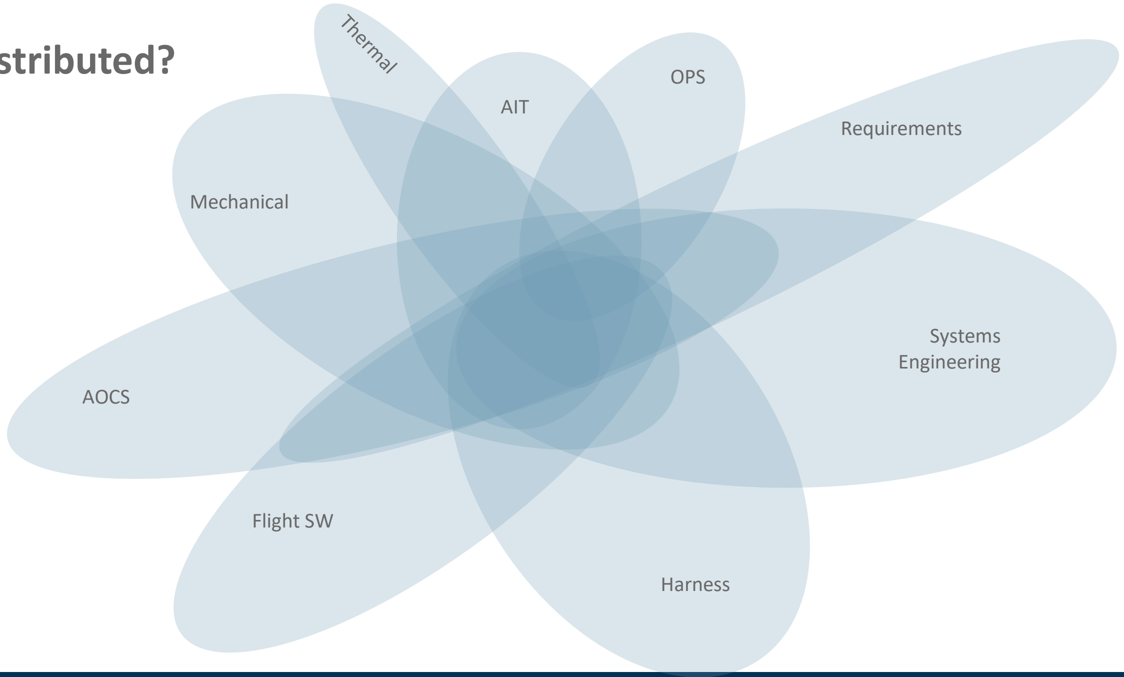
DATA HUB ARCHITECTURE TOPOLOGY

INTEGRATION OF SYSTEMS ENGINEERING AND OTHER DOMAIN-SPECIFIC TOOLS

How can we realize a distributed collaboration platform embracing many domains and domain specific tools?

Architecture Problem: **Where is the (shared) data stored?**

Workflow/Collaboration problem: **How is the (shared) data synchronized or distributed?**



DATA HUB ARCHITECTURE TOPOLOGY

INTEGRATION OF SYSTEMS ENGINEERING AND OTHER DOMAIN-SPECIFIC TOOLS

How can we realize a distributed collaboration platform embracing many domains and domain specific tools?

Architecture Problem: **Where is the (shared) data stored?**

Workflow/Collaboration problem: **How is the (shared) data synchronized or distributed?**

Many questions asked → not real answer, no (simple) solution at hand

Everybody 'feels' that we need to tackle this but

there is no answer (yet) but many more questions

QUESTIONS

CROSS DOMAIN MODEL-BASED ENGINEERING

There are different feasible architectures sketched before.

Are there any other architectures known, implemented, feasible or thought of?

- publisher/subscriber
  - Requires open API to exchange messages/data

    - standardized interface

    * Record the

- Point to point information handover with central recording of changes

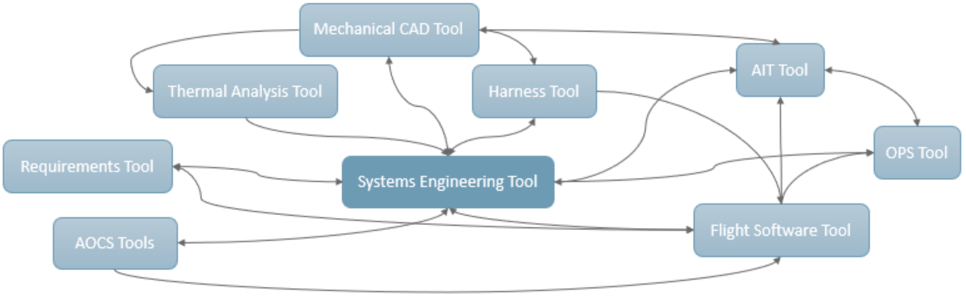
QUESTIONS

CROSS DOMAIN MODEL-BASED ENGINEERING

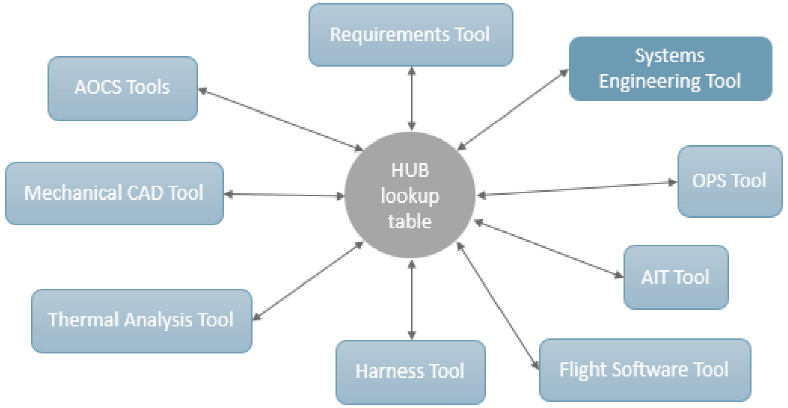
There are different feasible architectures sketched before.

Are there any other architectures known, implemented, feasible or thought of?

- **What is the decentralized/distributed/federated architecture?**



  - **Many point2point connections**
  - **Central lookup table / proxy / broker**



- Decentralized?
- Do we copy data or do we actually replicate data in different tools?
- Who is the owner of the data?

QUESTIONS

CROSS DOMAIN MODEL-BASED ENGINEERING

There are multiple different domain specific tools involved which need to align to efficiently cooperate

Which architecture (centralized vs federated) ?

- ”MBSE is centralized by definition” (single source of truth) …. Everything else is implementation detail
- it’s all about workflow (and the people involved)

  .. More important than the tooling topology
- Centralized approaches are likely to cause vendor lock-in

Way forward: (just like other engineering problems)

We need to put a set of use-cases on the table and …

… try to answer them in detail to find a good solution.

QUESTIONS

CROSS DOMAIN MODEL-BASED ENGINEERING

There are multiple different domain specific tools involved which need to align to efficiently cooperate

Which subjects need to be considered across multiple domain specific tools?

- **Semantic interoperability**
  - **A global semantics**
  - **Point-to-point sharing semantics / tool-specific mapping**
  - **Which one is actually feasible?**

    (maybe the global approach is too big to be successful ??)

QUESTIONS

CROSS DOMAIN MODEL-BASED ENGINEERING

Sharing data cross domains …

Share the data at “the right time” → version control / baseline ??

Share the data at the right maturity

Share not only the data, but add some meta-data,

    * like constraints (valid value range, margin .. maturity information ..)

    * that allows to continue working ‘on the island’

QUESTIONS

CROSS DOMAIN MODEL-BASED ENGINEERING

Version Control

- Depends a lot on the needs of the domain
- PDM/PLM systems are suitable for electrical/mechanical design
- Software requires git-like approaches

- Probably there is no ‘common’ approach on version control …
- Is that needed at all?
  - Maybe not
  - Version control con be done domain/tool – specific
  - Certain points to synchronize across domains

Model-based engineering data-hub architecture: centralised Vs decentralised

Group Presentation

MBSE 2023 Workshop
16/11/2023

Key Issues brought up during the discussion

- Configuration Management
- Data **ownership** (especially de-centralised approach)
- How to handle **security** –especially when classified data?
- How to manage **export control** –especially between different countries?
- When to freeze a model version?
- Who controls user access?
  - How?
  - When?

Key Outcomes

- → Tools cannot replace human communication –which is necessary for data exchange
- → How could tools help with decision making
  - Help with performing trade-offs
  - Trade-offs and final decision making is a human task after all
- → Ontology is crucial (data models etc.)
  - Need for clear data-mapping
  - Need for semantic adaptors (tooling)
- → Need to differentiate between various levels of formalisation of change e.g. versioning, baseline, iteration
- → Need for clear definition of inter-domain **links** & **interfaces**

Key Outcomes

- Need for (formal) definition of margins, **uncertainty levels & constraints** at each model iteration
- **Meta-data** as important as the data themselves
- Viewability & Readability of data hub is important → database format/style not adaptable
- Black-Box approach where everyone has access to some info but central control?

Centralised or De-Centralised?

- → Decentralised: more agile, faster exchange of info, easier access to data (free access) BUT security considerations, data governance + complexity of implementation; also depends on the domain
- → In favour of a more hybrid approach
  - Centralised data catalogue where all data is available (de-centralised approach)
  - But access is request-based
  - Systems Engineer should be the one deciding who has info to what etc. & taking final decisions
- → Ontology is crucial (data models etc.)
  - Definition of each element needs to be explicitly defined in order to allow sharing between domains