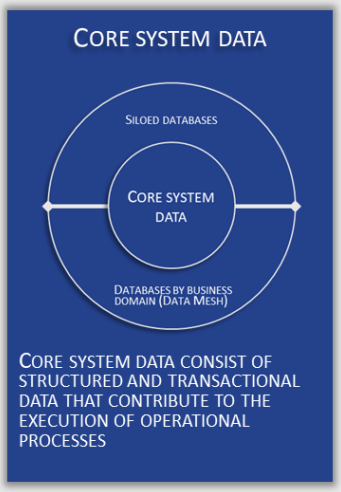
CORE SYSTEM DATA

Core system data consists of your structured and transactional data that contribute to the execution of operational processes, as well as links to unstructured and multimedia data structures. These data elements have predetermined usage objectives. This does not refer to decision-making system data (business intelligence, data analytics…). Core system data relies on OLTP technologies capable of handling high-frequency multi-user and multi-system concurrent access.



1. CONDITIONS OF SUCCESS

To successfully deploy AI at the enterprise level, it is essential to rely on an up-to-date understanding of the existing data within the information system, whether structured or unstructured. Mastering this knowledge ensures the improvement and maintenance of data quality, which is critical for the effectiveness of AI system training processes.

This understanding operates at the meta level. It is dedicated to describing existing data and their location within applications and data flows. It does not focus on data valorization, as addressed by an Operational Data Store (ODS) or Master Data Management (MDM) system. In other words, the objective is to document the intangible assets of the information system necessary for deploying AI across the enterprise.

In TRAIDA, the term "core system data" refers to this data repository that forms the operational core of the information system, covering both structured and unstructured (multimedia) data. All such data is described through metadata, which unifies their representation. Aggregated and statistical data are not a primary focus for documentation under the "core system data" framework. These are instead derived datasets located in layers further from the core of the information system (refer to TRAIDA Data Lake Warehouse card).

Lastly, the term "unstructured data" is used cautiously, as these datasets also have metadata that describe them, implying an underlying structure that needs to be documented for better reuse and updating. This is also the purpose of the data catalog described in this TRAIDA card.

Implement a data catalog.

If you do not have unified and up-to-date knowledge of your core-system data structures such as dataset names, table names, field names, relation names, you need to build or strengthen your metadata repository while avoiding extensive semantic modeling that could be lengthy and costly. This repository isn't meant to handle the data values but to help you understand the metadata managed in your core-system databases. It facilitates the creation of a business terms glossary that must be synchronized across all your operational systems.

This is a sort of data catalog, but it is limited to the work of capitalizing on the knowledge applied to core system databases. It does not replace a complete data catalog repository, which is usually managed through a Master Data Management (MDM) (see the related card).

The knowledge accumulated within the metadata repository highlighted in this card is essential to support and enhance your efforts in semantic modeling. This will provide the initial versions of the ontologies needed to increasingly scale your AI systems. More broadly, it will help you regain control of your data quality.

To achieve this goal, utilize graph-oriented database technology, which offers a schema-free approach for loading existing core system data along with their documentation and automatically computes an initial version of your metadata portfolio. This computation is driven by a generative AI (LLM) at the entry-point of the data injection. By combining agile graph technology with generative AI, you will quickly enhance your understanding of core-system data structures. You will apply a prompt similar to this one:

"Develop an ontology from the provided data repository, utilizing the initial list of business concepts, which you may further enrich. Ensure the removal of any duplicate concepts and clearly articulate the relationships between business concepts and existing elements, including applications, tables, fields, and relationships. The ontology should document all metadata, such as application names, table names, field names, and relationship names".

The result of this prompt is then used to generate the graph. AI prompting must be guided by your business terms glossary to create triples from every metadata item stemming from your core system to your official business terms. A triple consists of (1) a unified business concept, (2) a relation (linked to), and (3) an existing concept in your Information System, such as application and dataset names, table names, and field names.

The AI approach will enable a faster engineering process and avoid cumbersome modeling procedures that are inefficient due to rapidly changing data structures and complexity in your operational systems.

This knowledge enables you to determine actions to correct defects in existing silos, clean up data (meaning, completeness, accuracy, deduplication), optimize your data flow integration APIs (pivot format enhancement) and align better with regulations.

Additionally, you will accumulate the necessary knowledge to progressively redesign your siloed databases to organize them by business concepts (e.g. data mesh) and set up operational repositories such as your Operational Data Store (ODS, storage of data values) and Master Data Management (MDM, metadata repository at scale with data governance features applied to master and reference data).

Unstructured data

Your core system data have links to unstructured data sources, such as document management repositories or big data. The metadata catalog presented here should also take this into account. These are specific types of metadata that are considered "unstructured." Since this is not about storing the value of the data, the issue of storing these multimedia contents is not addressed here.

2. IMPORTANCE OF THIS CARD FOR YOUR TRANSFORMATIVE AI

The data from your core-system are necessary to train your AIs.

Static data (which does not change or changes infrequently) is used during mass training sessions (batch) and the more volatile data is used to refine queries (prompt injections, RAG technology, etc.) in real time (on the fly). To facilitate the implementation of unified data repositories necessary for AI systems, such as Operational Data Store (ODS), Master Data Management (MDM), or Enterprise Knowledge Graph (EKG), you must have good documentation of your core-system data structures and meanings. To achieve this goal, it is beneficial to set up a metadata catalog that describes them in synchronization with the MDM system.

EXISTING SILOED DB

Depending on the quality of your existing core system databases, you will be more or less well-prepared for the deployment of an enterprise-wide data governance policy with the right foundation for your AI systems.

If you encounter obstacles when deploying new transactional data and when developing processes that extend across multiple databases, then consider the success conditions described in this card. It is unlikely that you can significantly improve your situation through a single data governance initiative or data mesh project alone. Indeed, having up-to-date and detailed knowledge of your data structures (metadata) is a prerequisite for any improvement actions.

As indicated in this card, you need to build this knowledge quickly, cost-effectively, and so that it can be easily updated as your core-system evolve. This objective is addressed by the use of graph-oriented database technology in schema-free mode coupled with generative AI.

BUSINESS DOMAIN DB WITH DATA MESH

Data mesh is a data architecture approach. Its objective is to organize databases around business concepts, as opposed to organizational and functional silos. It's an "object-oriented" approach at a systemic data scale.

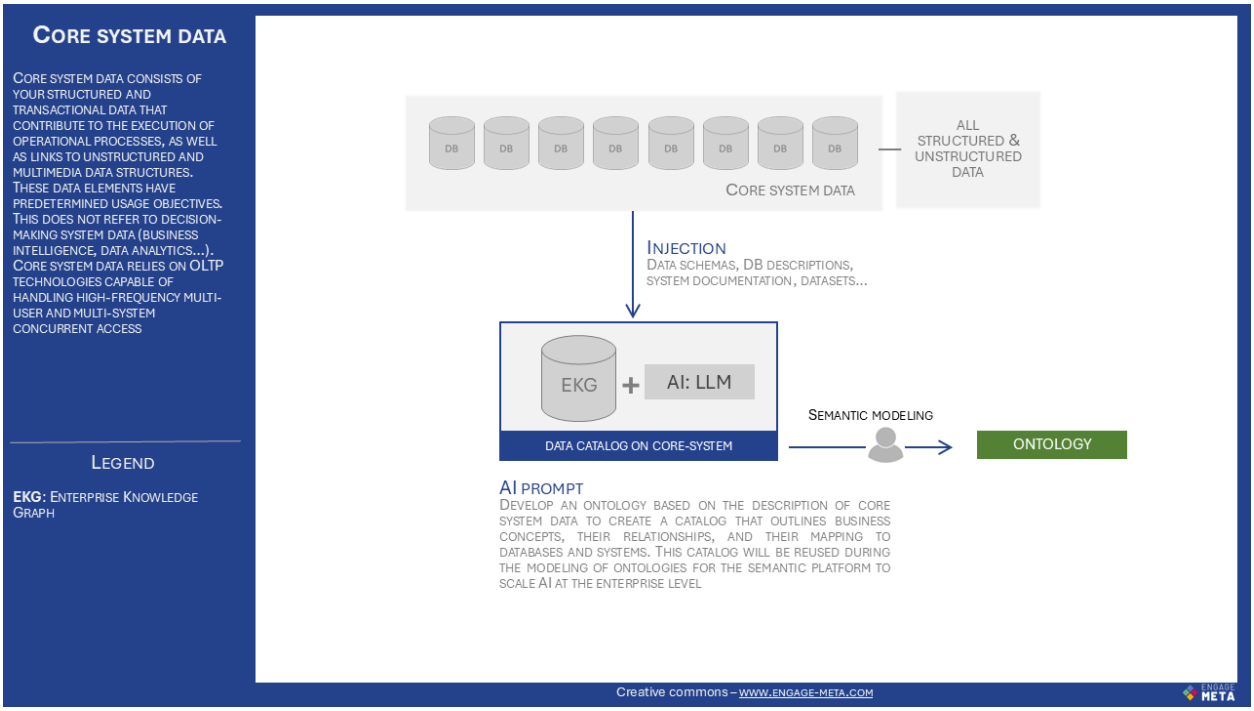
With a database for each business concept, the issues of data duplication across silos are eliminated. Operational processes then source their data through standardized access in each database of the involved business concepts.

To implement this approach, you must act cautiously to avoid malfunctions at these levels:

- management of links between data located in different databases in terms of integrity, transaction management, and data flows,
- normalization of metadata that must be common to all databases,
- service level agreement on response times and concurrent accesses,
- unified data preservation in histories and archives,
- engineering of software development that exploits these new databases and security.

To succeed in a data mesh program, first, ensure sufficient and up-to-date knowledge of the existing core-system databases structures (metadata), which we have already discussed in the success conditions of this card.

3. BLUEPRINT



4. YOUR SITUATION & OBJECTIVES