SAND2024-05188C

SANDIA REPORT:

Sandia National Laboratories

Digital Engineering

Product Realization Integrated Digital Enterprise (PRIDE)

GUIDE TO SELECTING A DATA INTEGRATION FRAMEWORK

Peter E Chandler

Digital Engineering Conference 2024
Idaho National Laboratory
April 29 – May 1, 2024

Uncontrolled Unclassified Information (UUI) / Unclassified Unlimited Release (UUR)

U.S. DEPARTMENT OF ENERGY NNSA

Sandia National Laboratories is a multimission laboratory managed and operated by National Technology and Engineering Solutions of Sandia, LLC., a wholly owned subsidiary of Honeywell International, Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.

Digital Engineering Decomposed

Digital Engineering is an **integrated digital approach** that uses **authoritative sources of system data and models** as a **continuum across disciplines** to support **lifecycle** activities from concept through disposal

Enterprise Integration

Site-specific Integration

Program Integration

INTEGRATION ACROSS DISCIPLINES, LIFECYCLE

Descriptive Modeling

Analytical Modeling

Visualization

DIGITAL APPROACHES

Design Sources

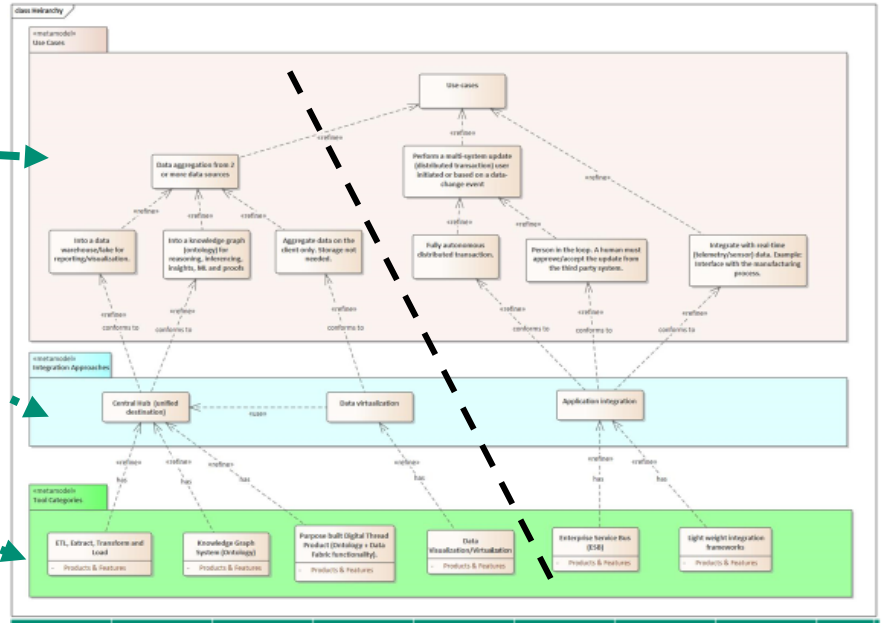
Production Sources

Programmatic Sources

SOURCES OF SYSTEM DATA

AGENDA

- Common Integration Use Cases
- Integration Approaches
- Tool Category
- Products



Tool Categories	Do nothing	ETL and Data Warehouse	ESB	Light weight integration frameworks	Data virtualization	Knowledge Graph System (Ontology)	Purpose built Digital Thread Product	Build your own
Companies Products	N/A	Abe Initio Integrate.io Talend Open Studio for Data Integration Informatica Power Center Oracle Data Integrator Denodo Platform Microsoft SQL Server Integration Services Pentaho Data Integration IBM DataStage Hevo Data	Workday Business Process Framework, Akule ESB, WebSphere Message Broker (discontinued), Oracle Service Bus, IBM DataPower Gateway, WSO2 Enterprise Service Bus, Azure Service Bus, Red Hat Fuse, Talend Enterprise Service Bus, and Fiorano ESB, eQube	Apache Camel, Spring Integration	Denodo Informatica PowerCenter Oracle Data Service Integrator TIBCO Data Virtualization	Apache Jena, GraphDB, RDF4J, MarkLogic, Virtuoso, AnzoGraph, Stardog	SBE Vision, DeepLynx, k2view, Stardog, Talend Data Fabric, Syndeia™ digital thread platform	N/A

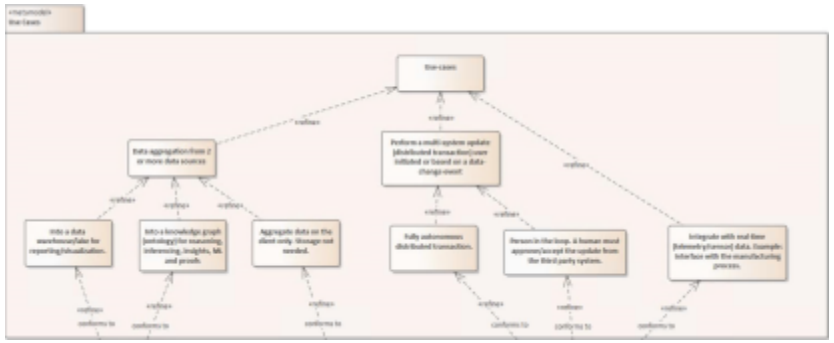
Common Use Cases

Backoffice Analysis

- Data Aggregation (read) from Two or More Data Sources
  - Into a data warehouse/lake for reporting/visualization.
  - Into a knowledge graph (ontology) for reasoning, inferencing, insights, machine learning and proofs
  - Aggregate data on the client only. Storage not needed.

Operational

- Perform a Multi-system Update (Distributed Transaction) User Initiated or Based on a Data-change event
  - Fully autonomous distributed transaction.
  - Person in the loop. A human must approve/accept the update from the third-party system.
- Integrate with Real-time (Telemetry/Sensor/IoT) Data. Example: Interface with the Manufacturing Process



Integration Approaches

Data Virtualization

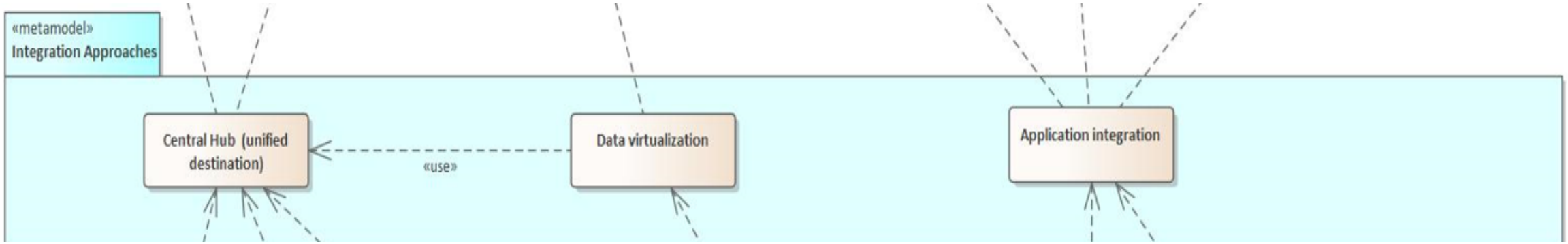
- software acts as a bridge across multiple, diverse data sources.

Application Integration

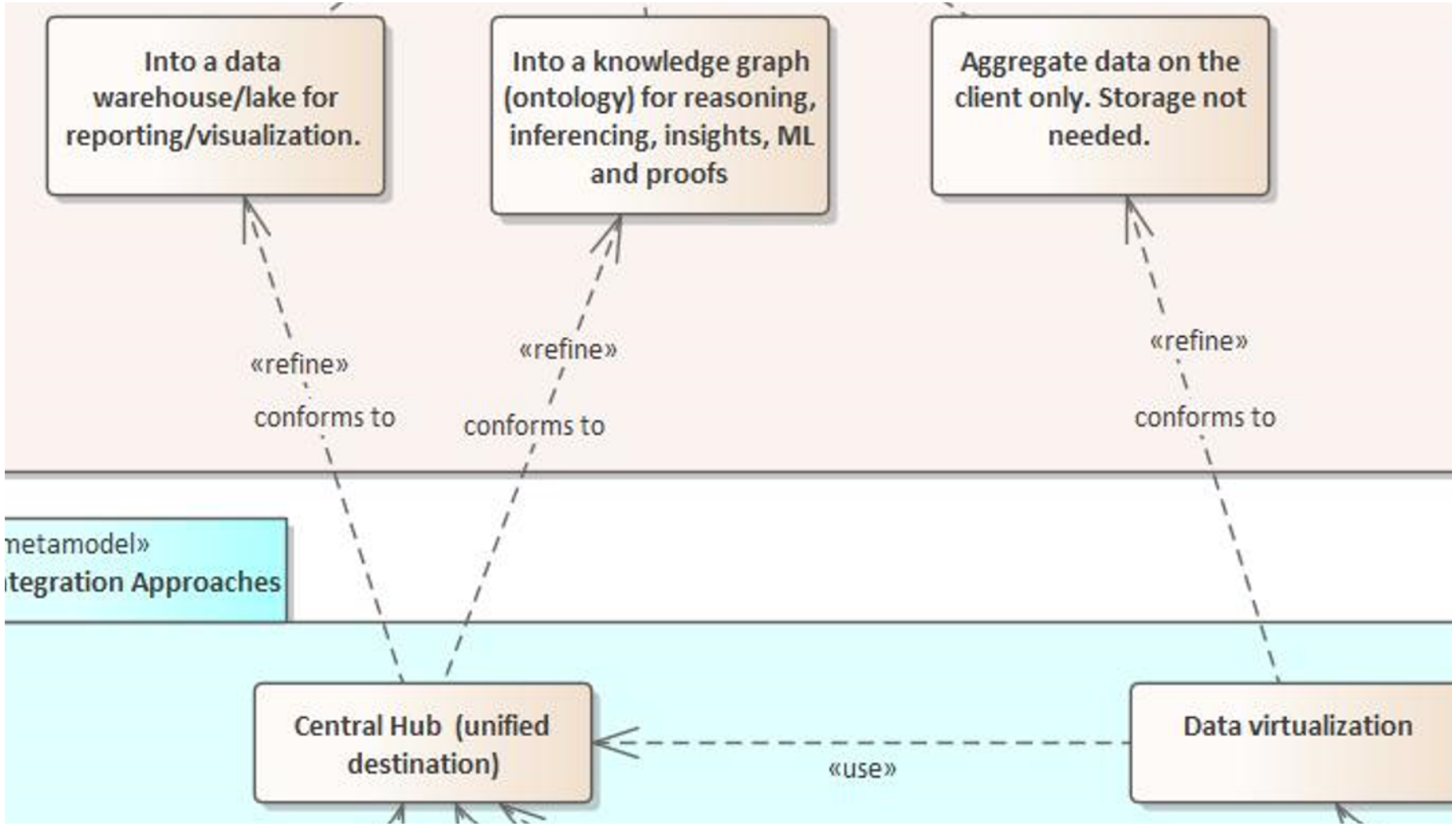
- the process of enabling individual systems and applications, each designed for its own specific purpose, to work with one another. Aka composite application.

Central Hub (Unified Destination)

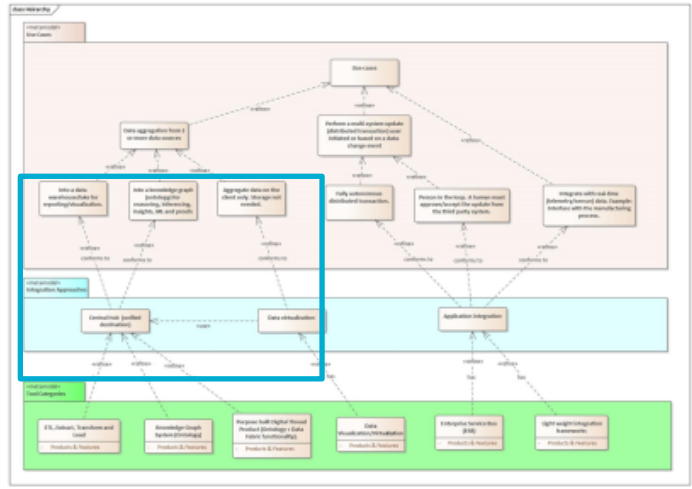
- sending data from across an enterprise/network into a centralized system such as a database, a data warehouse, a data lake, triple/quad store, or a data lake-house.



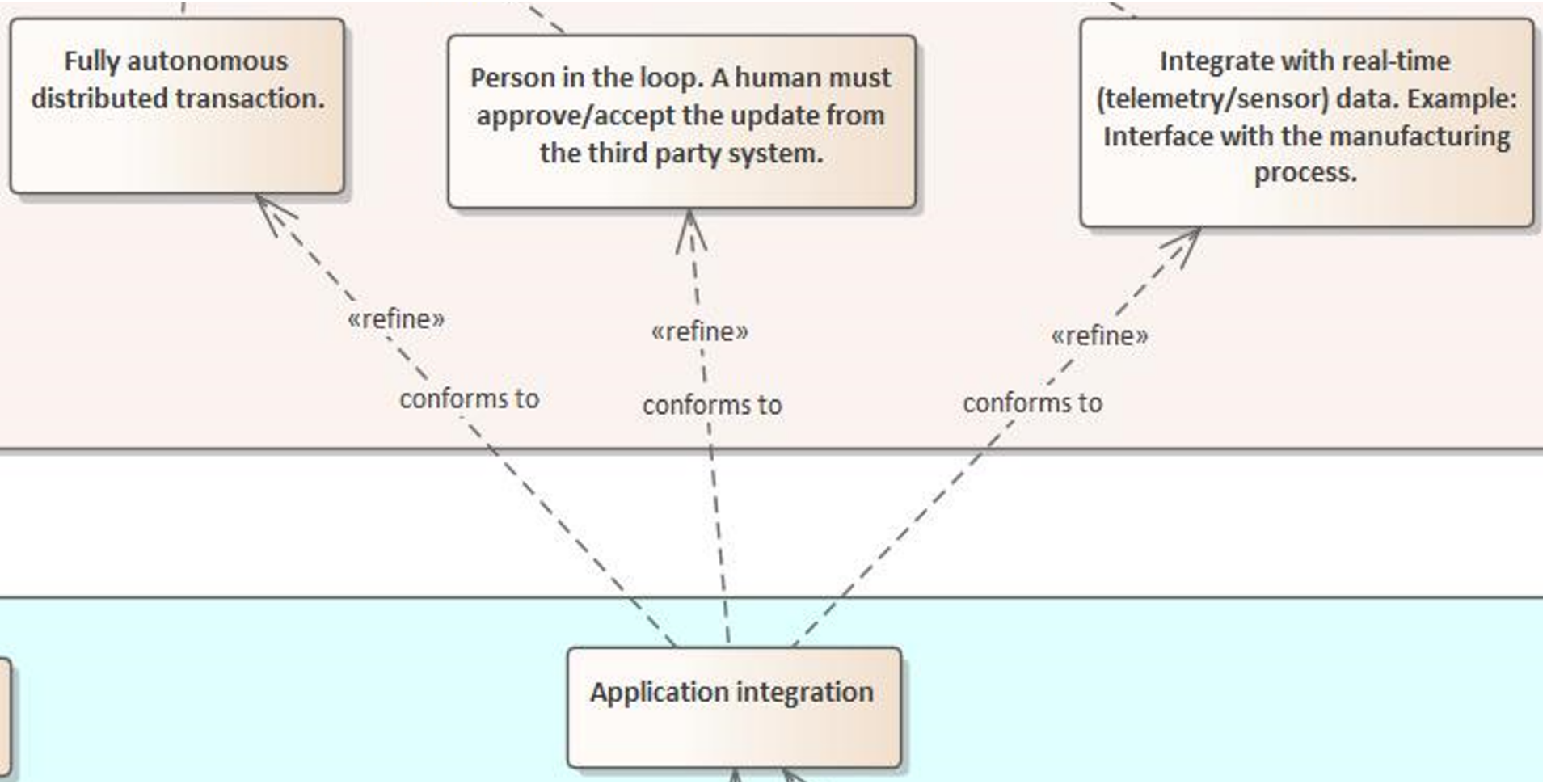
Data Aggregation (read) from Two or More Data Sources



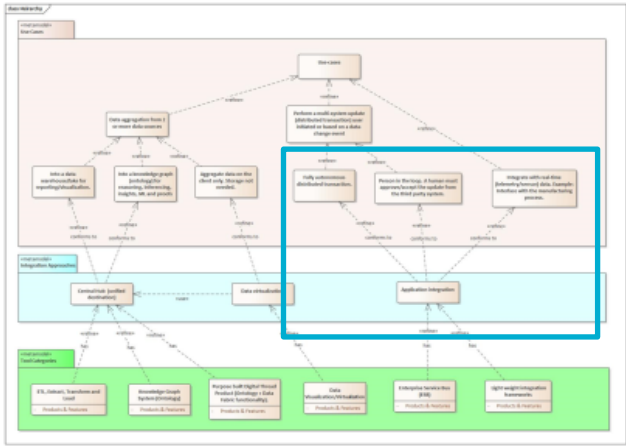
Use-cases mapped to *Integration Approaches*



Perform a Multi-system Update (Distributed Transaction) User Initiated or Based on a Data-change event



Use-cases mapped to *Integration Approaches*



Tool Category

Do Nothing/Ad Hoc

- Currently, we have many point-to-point integrations creating integration “spaghetti.”

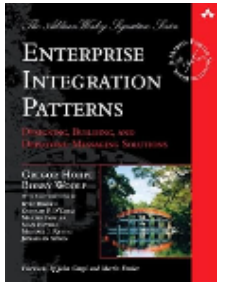
ETL/ELT and Data Warehouse

Enterprise Service Bus

- Is an architectural pattern whereby a centralized software component performs integrations between applications.

Lightweight Integration Frameworks

- Lightweight integration frameworks implemented Enterprise Integration Patterns EIP



Data Virtualization/visualization

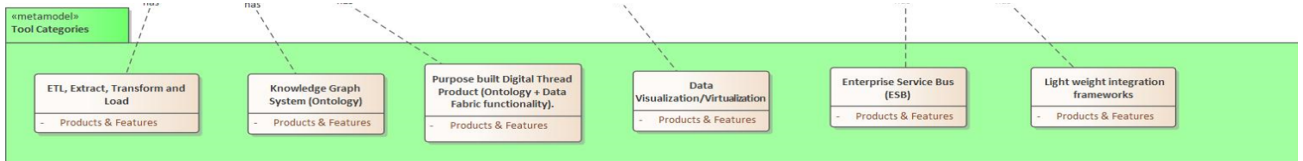
Knowledge Graph System (Ontology)

- Ontologies are stores using a semantic web, resource description framework, triple store.

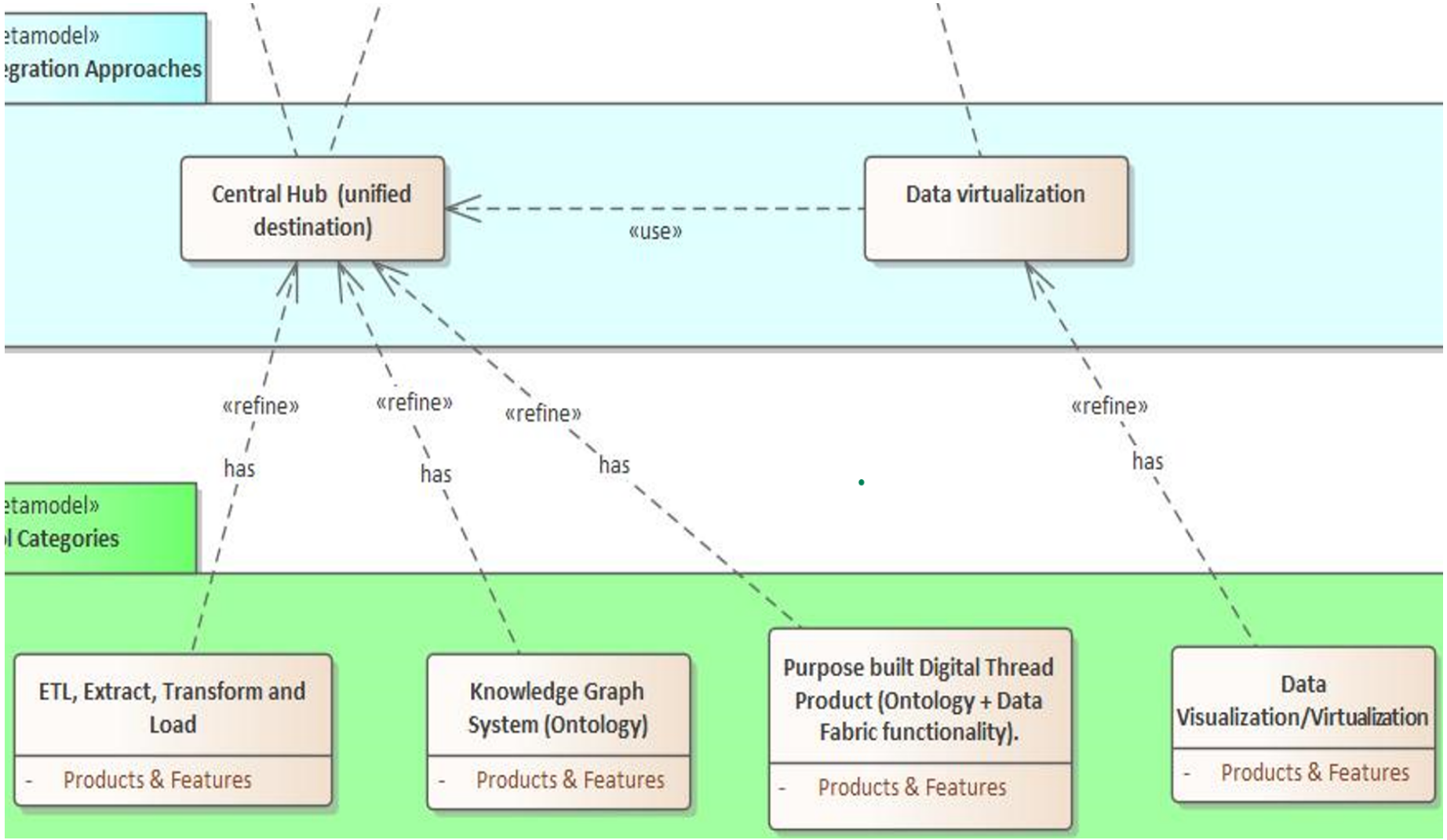
Purpose built Digital Thread Product (Ontology-based Data Fabric)

- Ontology + Data-Fabric: are emerging technologies.

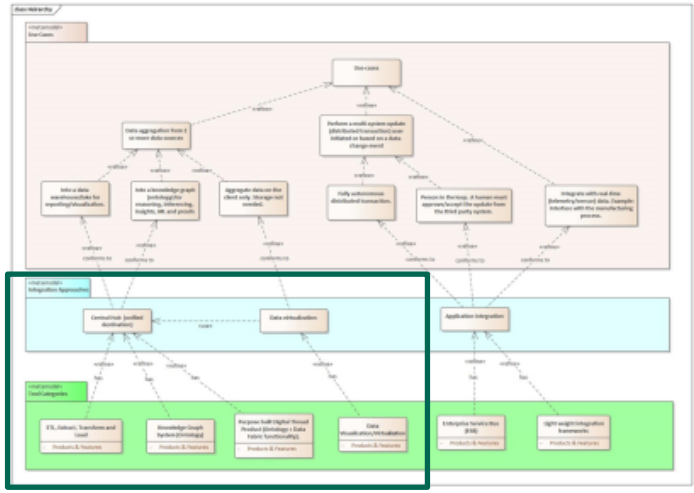
Build Your Own



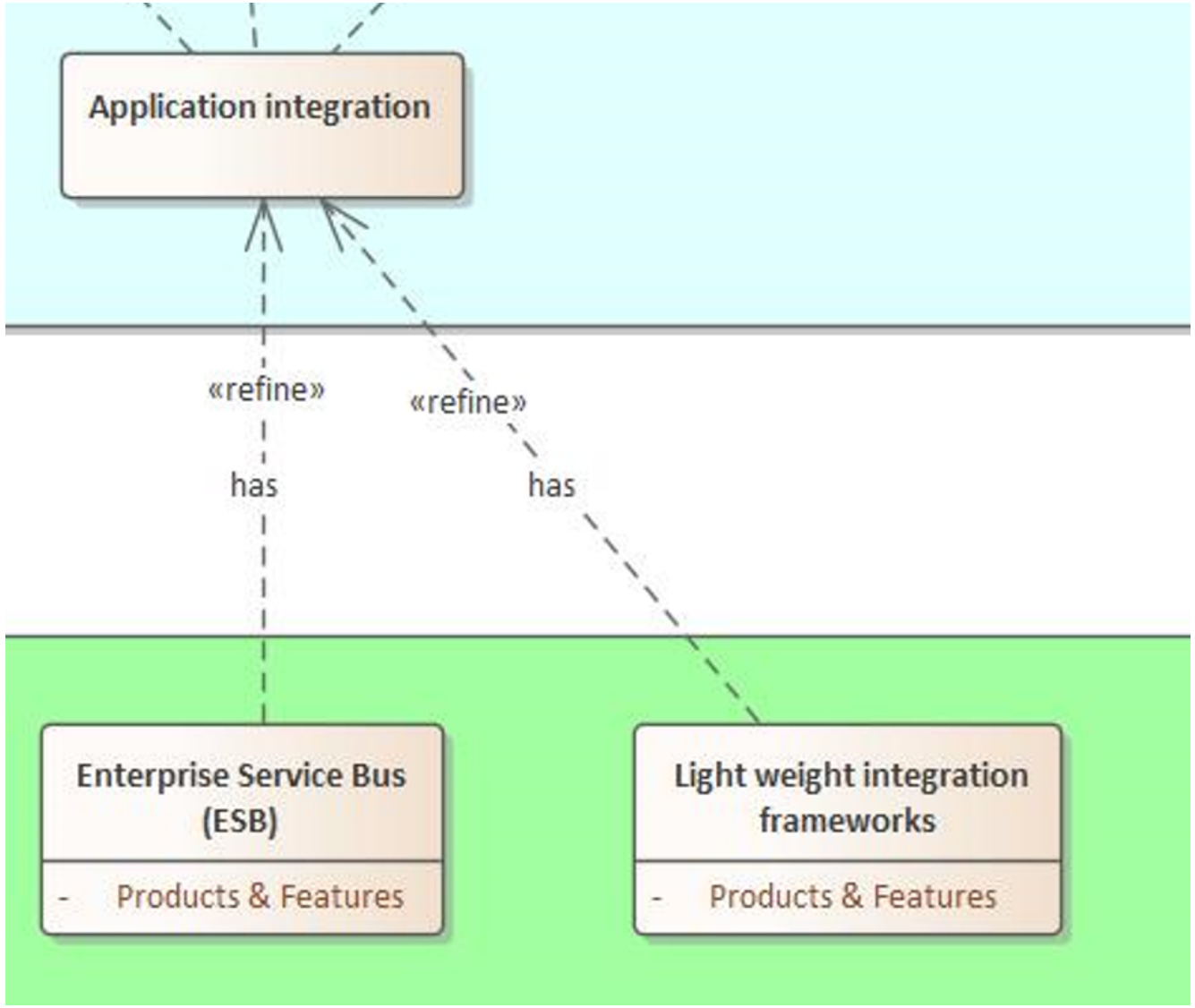
Data Aggregation (read) from Two or More Data Sources



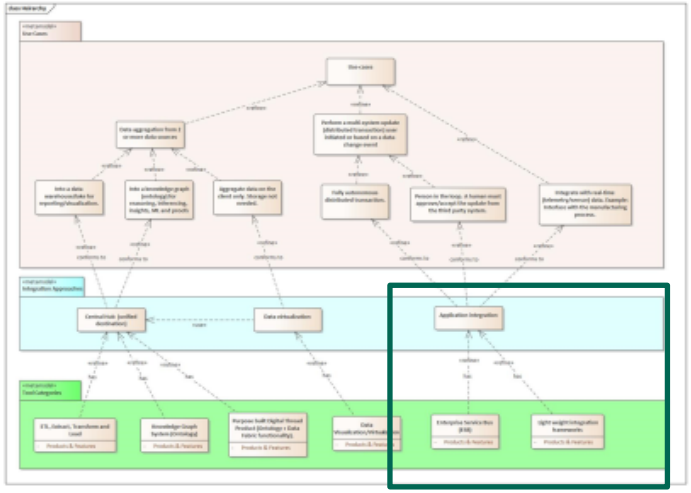
Data Aggregation (read) mapped to *Tool Category*



Application Integration



Application Integration (update, create, delete) mapped to *Tool Category*



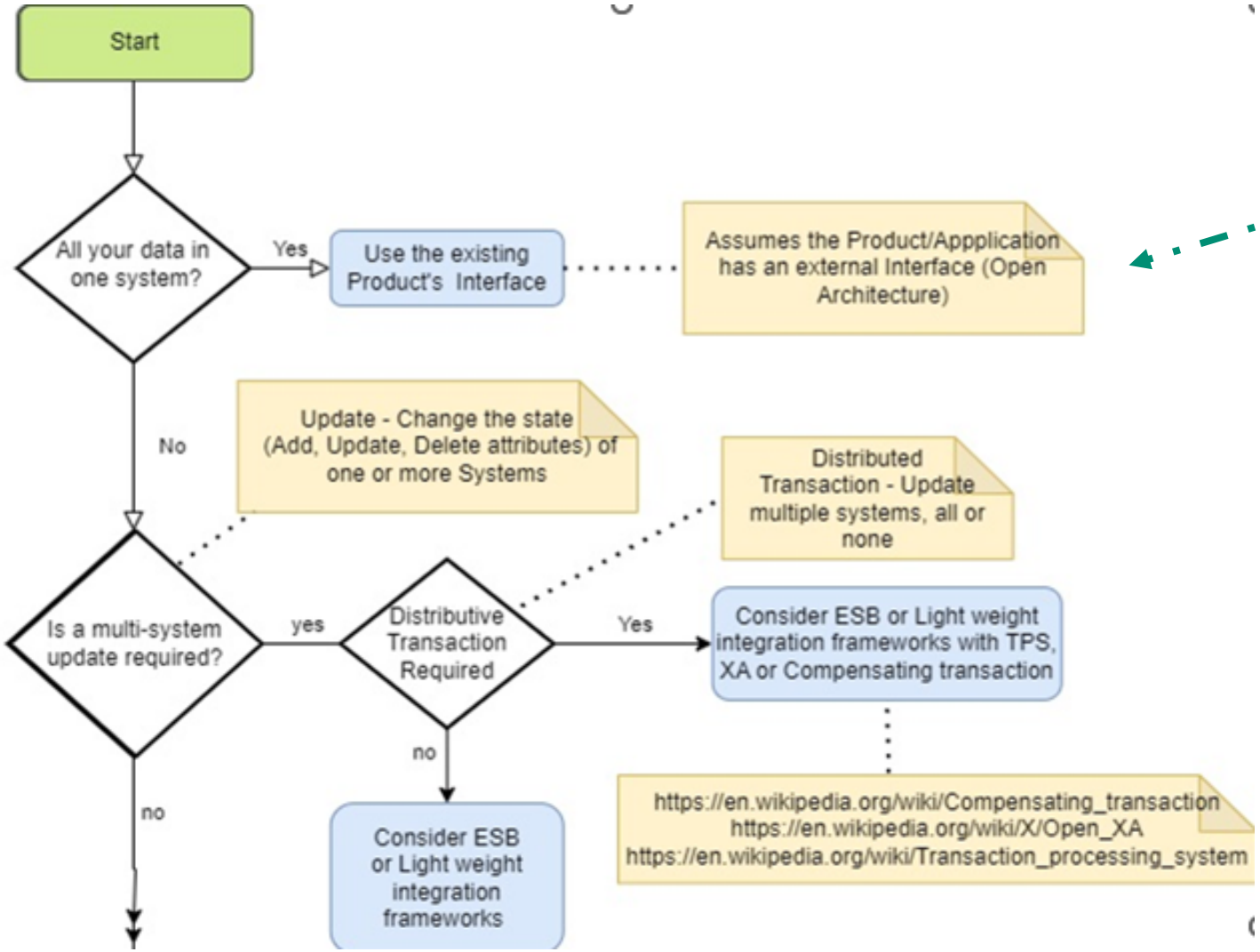
Sample Products per Tool Category

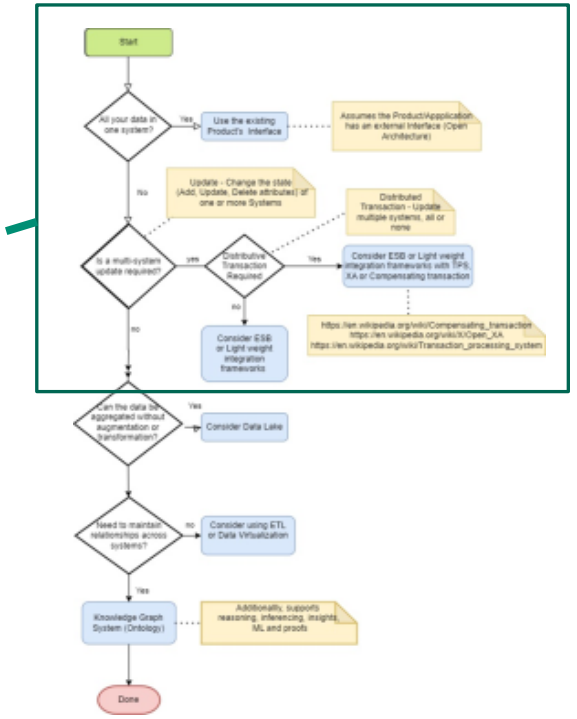
Tool Categories	Do nothing	ETL and Data Warehouse	ESB	Light weight integration frameworks	Data virtualization	Knowledge Graph System (Ontology)	Purpose built Digital Thread Product	Build your own
Companies Products	N/A	Abe Initio, Integrate.io, Talend Open Studio for Data Integration Informatica Power Center Oracle Data Integrator Denodo Platform Microsoft SQL Server Integration Services Pentaho Data Integration IBM DataStage Hevo Data	Workday Business Process Framework, Mule ESB, WebSphere Message Broker (discontinued), Oracle Service Bus, IBM DataPower Gateway, WSO2 Enterprise Service Bus, Azure Service Bus, Red Hat Fuse, Talend Enterprise Service Bus, and Fiorano ESB, eQube	Apache Camel, Spring Integration	Denodo, Informatica, PowerCenter, Oracle Data Service Integrator, TIBCO Data Virtualization	Apache Jena, GraphDB, RDF4J, MarkLogic, Virtuoso, AnzoGraph, Stardog	SBE Vision, DeepLynx, k2view, Stardog, Talend Data Fabric, Syndeia™ digital thread platform	N/A

In this table we added definitions and sample products. The sample products are not a complete list but a sampling within the tool category.

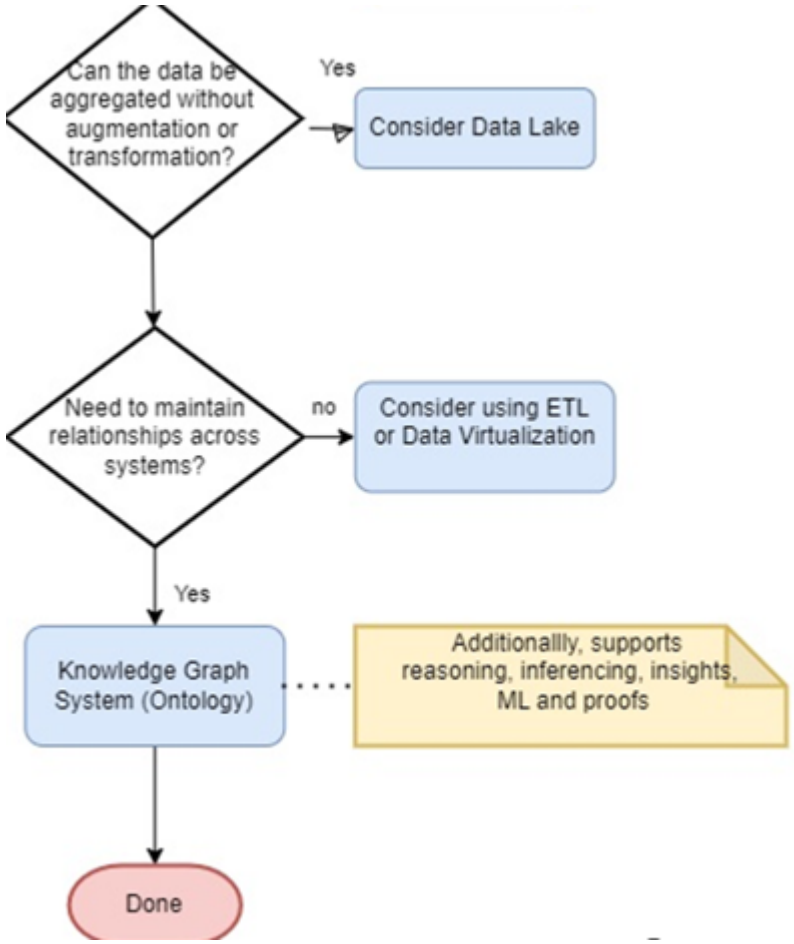
Thank You
Q&A Time!

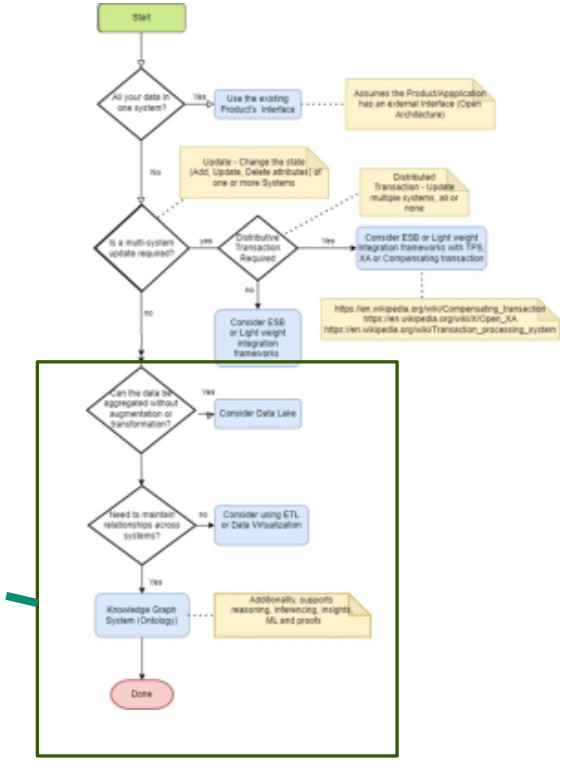
Category Selection Flow Chart 1/2





Category Selection Flow Chart 2/2





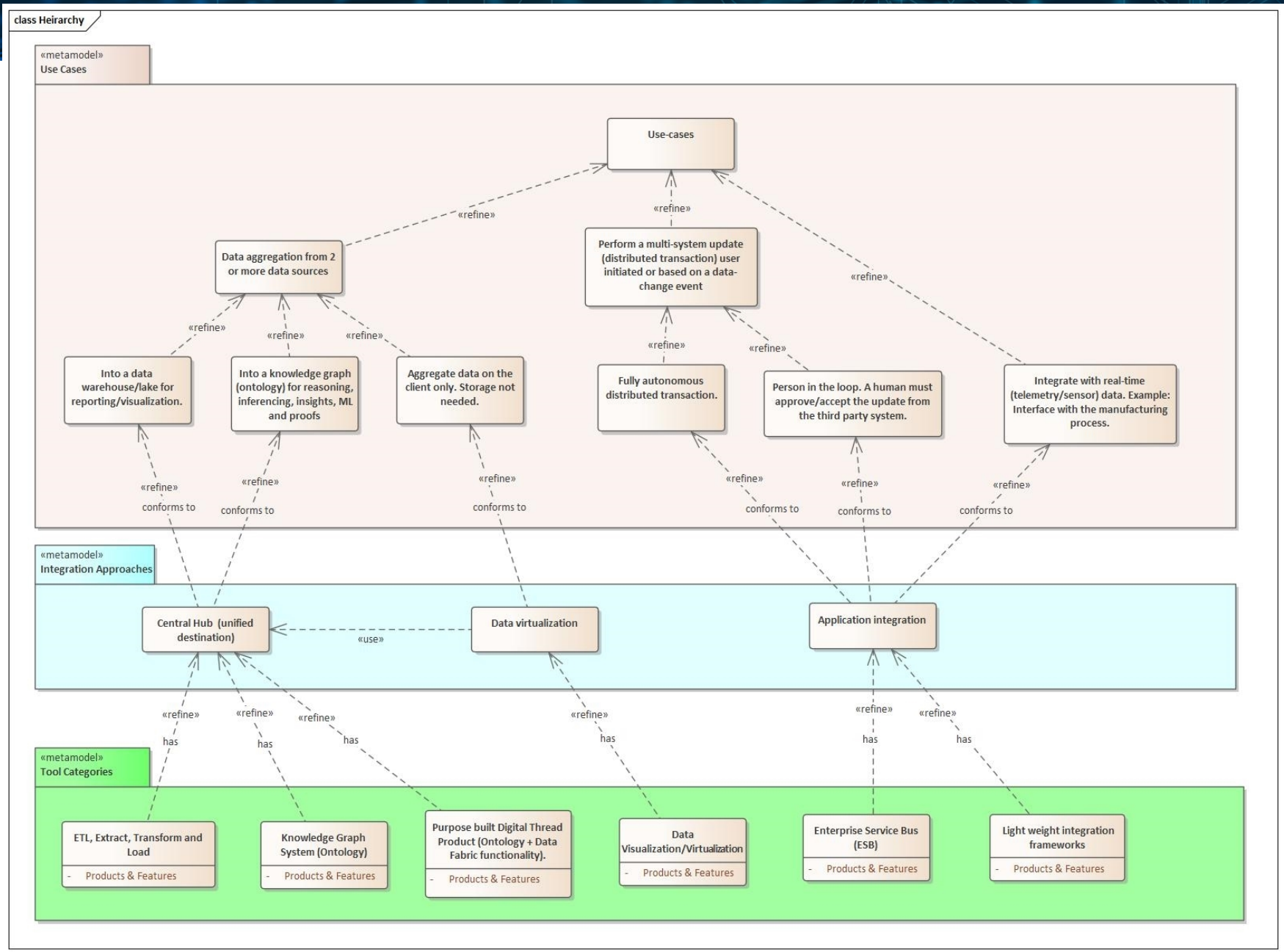
INTEGRATION MANIFESTO (BEST PRACTICES/PRINCIPLE) 1/2

1. No one domain, organization, or application owns data. The Parent Organization is the data owner. i.e., "Free the data." All domains are responsible for sharing data with other need-to-know domains.
2. Avoid creating more static copies of the data. Example: warehouses, repository, or DataMart.
3. Data persists within a Domain and its applications. The tool or application that creates the data should store it when possible.
4. The "system of record," i.e., truth for data (authoritative source) can vary, as it is a function of the product development phase. There should be one and only one "system of record" at a given time.
5. Applications (COTS, GOTS, custom-developed) should provide a standard based external (network) interface. Some standards to be considered are: REST, SOAP, OpenAPI, OSLC, RFD, ISO/IEC JTC 1/SC 7, STEP AP 239, FMI, XMI
6. All new integrations should be M x N, i.e.; multiple different applications can pull data and push data. Consider Publish/Subscribe pattern. Avoid point-to-point solutions.

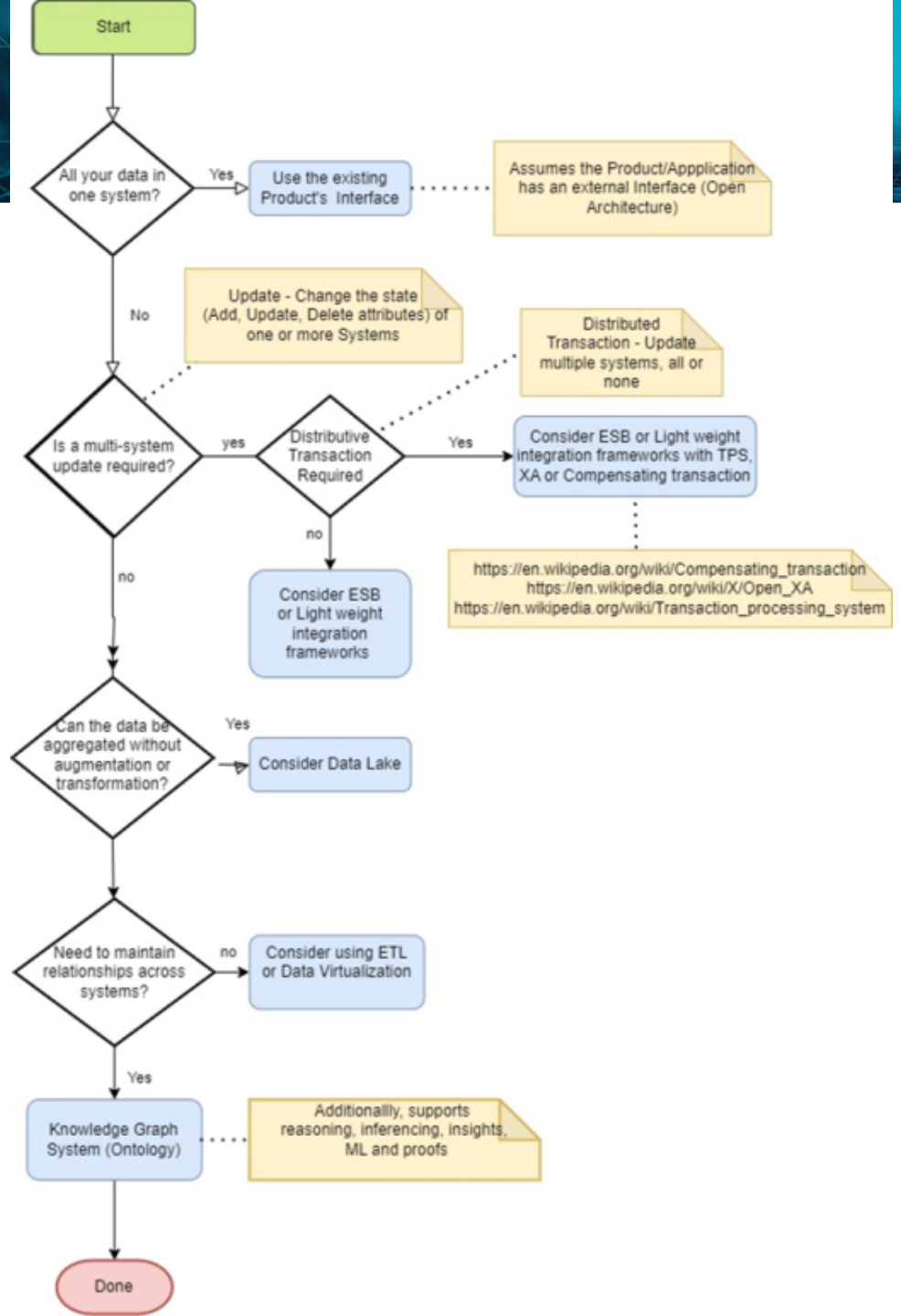
INTEGRATION MANIFESTO (BEST PRACTICES/PRINCIPLE) 2/2

7. All data should be in a neutral/universal/canonical form that is self-describing. Consider the RDF format for application data.
8. Enterprise Integration Patterns EIPs should be used whenever possible.
9. When data is published or consumed, it should be validated (cleaned).
10. Data transfer patterns should be defined: on-demand (synchronous), scheduled, and event-based (asynchronous).
11. Distributed transactions require integrations compensating transactions or XA support.
12. Information transfer should be monitored, have tracing, and failure detection.
13. Avoid database-level integration. Use the vendor's API
14. Integrations should be versioned

Use Cases, Integration Approach, and Tool Category



Important Selection Questions



Access the full FY23 PRIDE Analysis of Alternatives, Data Integration Framework Report, contact:

Email: wg-Amalgamators-Team@sandia.gov