Integration Theory 101:
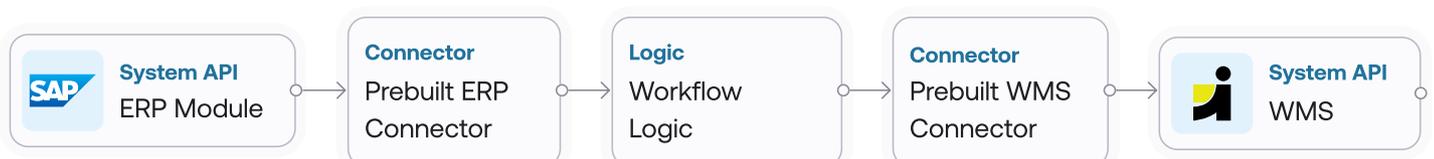## *Terms And Definitions*

Foundational Concepts

1. **Source System:** Where the data originates from.

2. **Target System:** Where the data travels to.

3. **Event:** A change in the source system (eg. new order created).

4. **Trigger:** The thing that starts the integration, often a schedule or event.

5. **Action:** What the integration does after the trigger (eg update order in system B).

6. **Workflow:** A complete process of data moving between system A and system B.

7. **Deploy:** Transferring your project from a staging environment to a live-production environment.

Data & Transformation Concepts

1. **Schema:** The structure/ shape of data (field/ type/ format).

2. **Mapping:** Connecting field in one system to another.

3. **Transformation:** Changing data as it moves between systems (formatting and logic).

4. **Payload:** The actual data being sent (eg, JSON).

5. **Data Validation:** Ensuring the data being sent is correct and meets the requirements of the target system.

6. **Custom Logic:** User-defined rules or transformations that dictate how data is processed or actions are executed within a workflow.

7. **Webhook:** Allow two systems to communicate with each other in real-time by automatically sending data when a specific event occurs.

The Basic Anatomy of An Integration

| **System API**
ERP Module | → | **Connector**
Prebuilt ERP Connector | → | **Logic**
Workflow Logic | → | **Connector**
Prebuilt WMS Connector | → | **System API**
WMS |

Flow Logic Concepts

1. **One-Way Sync:** Data flows from system A to system B.

2. **Target System:** Where the data travels to.

3. **Bi-Directional/ Two Way Sync:** Data flows A → B and B → A, usually with conflict rules.

4. **Event Driven Flow:** Triggered by changes happening in real-time (webhooks).

5. **Schedule/ Polling:** Runs at intervals (e.g., every 15 minutes).

6. **Routing/ Conditional Logic:** Rules that determine what goes where (if X, then send to Y).

Integration Technology Concepts:

1. **API:** (Application Programming Interface) How systems talk to each other.

2. **Authentication / Auth:** How systems verify your access (OAuth, API keys).

3. **Connector / Integration Connector:** A prebuilt component for connecting to a particular API that comes with a pre-configured understanding of its fields and API functionality EG rate limits.

4. **Embedded Connectors:** Handle the authentication, communication, and data exchange with a specific platform or API.

5. **Connector Engine:** A component of the Versori platform that allows us to dynamically spin up connectors based on API documentation. Additionally, these can be customised based on a cURL request into the target platform to capture custom fields or objects.

6. **Endpoint:** The specific API address where data is sent/received.

7. **Rate Limiting:** The maximum number of requests an API allows.

Error & Quality Concepts:

1. **Error Handling:** How an integration behaves when something goes wrong.

2. **Retry Logic:** What happens when a request fails.

3. **Logging:** Records of what the integration did (requests, responses, failures).

4. **Monitoring / Observability:** How customers can see the state of their integrations.

5. **Throughput:** How many records/events your integration processes over time.

6. Edge Case: A problem, input, or situation that occurs at the extreme boundaries or limits of the system's normal operating parameters.

7. **Data Integrity:** Ensuring the data stays accurate during the integration.

8. **Error Handling:** Clear, predictable behaviour for handling unexpected situations.

9. **Observability:** The ability to see what an integration is doing, including: logs, run history, error reports, data previews, success/failure rates.

Artificial Intelligence (AI) Concepts

1. **Agent:** An AI-driven component that can take actions, make decisions, or perform tasks based on instructions and context.

2. **MCP (Model Context Protocol):** Central platform module that manages, monitors, and coordinates agents, workflows, and integrations.

3. **RAG (Retrieval Augmented Generation):** AI approach that enriches generated outputs with information retrieved from external sources or knowledge bases

4. **Prompt:** The input or instruction given to an AI model. Good prompts lead to better outputs.

5. **Context Window:** The amount of information an AI model can consider at one time (like short-term memory).

6. **Vector Database:** A specialised database that stores embeddings and enables fast semantic search, commonly used in RAG workflows.

7. **Hallucination:** When an AI model generates information that sounds plausible but is incorrect or made-up. RAG reduces this.

8. **Reasoning:** How an AI organises steps to solve a problem. Some workflows rely on controlling or guiding this.