[DOCUMENT: PHASE 3 ARCHITECTURAL BRS]

Project: Catalyst AI Innovations — Enterprise Integration Layer

Status: Architectural Draft

Classification: Confidential / Technical

1. **Functional Objective: The Data Commit Loop**

   The primary goal of Phase 3 is to transition from Static Data Sanitization (Phase 1 & 2) to Dynamic System Writing. This requires the "Agentic Workforce" to not only read and clean data but to commit that data into the host's Core Insurance Platform (CIP) with 100% ACID compliance (Atomicity, Consistency, Isolation, Durability).

2. **Integration Topology: The "Connector" Agent**

   We utilize a Non-Invasive Integration Strategy. Instead of high-risk database surgery, we deploy "Connector Agents" that interface via:

   - **RESTful API Wrappers:** Utilizing the host system's existing endpoints.
   - **Direct Database Staging:** Writing to a secure "Shadow Table" before a human-approved trigger pushes data to the production environment.
   - **Headless Interface Automation:** For legacy systems lacking APIs, the agent utilizes secure, backend-driven navigation to commit data via the UI layer.

3. **Logical Mapping & Transformation (The "DASA" Handshake)**

   The BRS defines the "Handshake" between the Heuristic Engine and the Target Schema:

   - **Field Alignment:** Mapping "Member Name" from an unstructured Excel file to the exact database field (e.g., tbl_PolicyMembers.Member_FirstName).
   - **Type Enforcement:** Ensuring ZAR (R) values are converted to the correct decimal precision required by the insurer's financial module.
   - **Identity Resolution:** Using the Luhn Algorithm agent to verify South African ID numbers before they are written to the database to prevent duplicate record creation.

4. Security & Compliance Protocols

To satisfy POPIA and SOC 2 requirements during the data write process:

- **Encryption at Rest & Transit:** All data packets moving between the Catalyst Engine and the Host Database are encrypted using AES-256.
- **Immutable Audit Logs:** Every "write" command executed by an agent is logged with a timestamp, the specific Agent ID, and a reference to the HITL (Human-in-the-Loop) approval code.
- **Zero-Persistence Data Handling:** Once the commit is confirmed, the Catalyst Engine purges the raw PII (Personally Identifiable Information) from its temporary cache.