Metro Ticket Generating System in ServiceNow

Data Architecture

1. Description

Data Architecture defines how metro ticket data is collected, stored, managed, and utilized within the ServiceNow platform. It ensures structured data handling, integrity, and scalability to support automated ticket generation and reporting.

2. Architecture Overview

The system uses custom tables and catalog-driven data capture to maintain structured and centralized ticket information. Data flows seamlessly from the Service Catalog to backend tables through automated workflows.

3. Key Components

- **Custom Table: Metro Station's Details**
    - Stores station-related information
    - Used for source and destination mapping

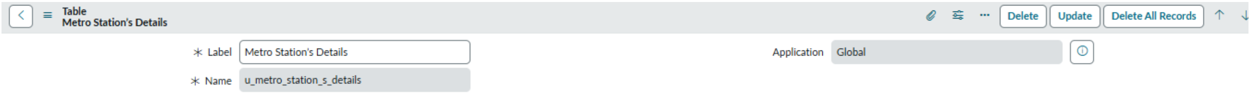


Table
Metro Station's Details

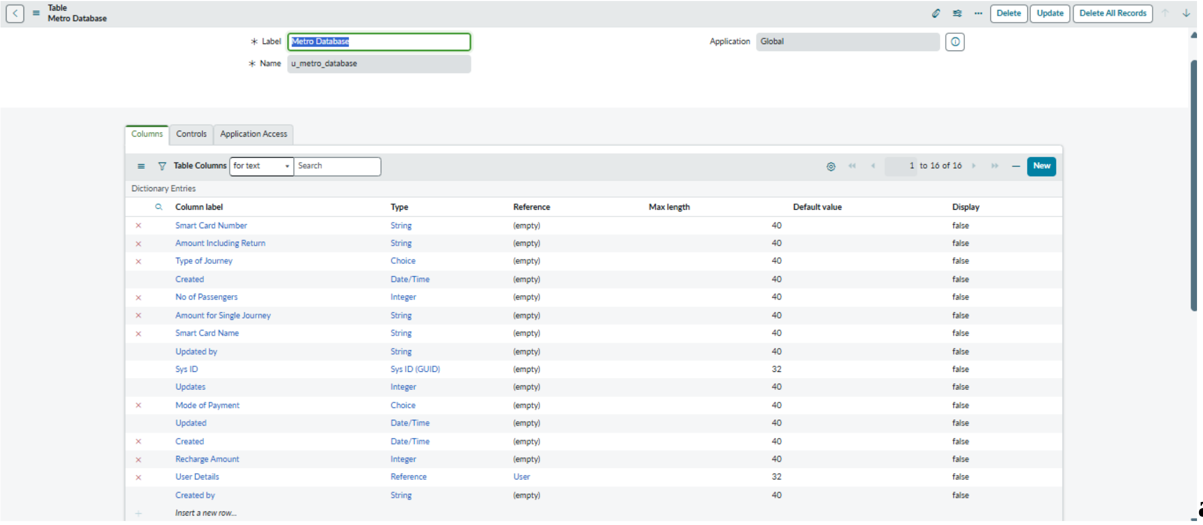
* Label: Metro Station's Details

* Name: u_metro_station_s_details

Application: Global

Figure 3.1: Creation of Table – Metro Station's Details

- **Metro Database Table**
    - Stores ticket booking details
    - Captures passenger, journey, and fare data



Label: Metro Database

Name: u_metro_database

Application: Global

Dictionary Entries

Column label	Type	Reference	Max length	Default value	Display
Smart Card Number	String	(empty)	40		false
Amount Including Return	String	(empty)	40		false
Type of Journey	Choice	(empty)	40		false
Created	Date/Time	(empty)	40		false
No of Passengers	Integer	(empty)	40		false
Amount for Single Journey	String	(empty)	40		false
Smart Card Name	String	(empty)	40		false
Updated by	String	(empty)	40		false
Sys ID	Sys ID (GUID)	(empty)	32		false
Updates	Integer	(empty)	40		false
Mode of Payment	Choice	(empty)	40		false
Updated	Date/Time	(empty)	40		false
Created	Date/Time	(empty)	40		false
Recharge Amount	Integer	(empty)	40		false
User Details	Reference	User	32		false
Created by	String	(empty)	40		false

Figure 3.2: Metro Database Table Structure

- **Catalog Variables**
  - Act as data entry points for users
  - Automatically mapped to backend tables

4. Data Flow

1. User submits ticket request via Service Catalog
2. Catalog variables capture journey details
3. Flow Designer maps variables to Metro Database
4. Records are stored for tracking and reporting

5. Benefits

- Centralized data storage
- Easy reporting and auditing
- Scalable architecture for future enhancements