# Feed Monitoring System - Pseudocode and Execution Plan

## 1. Approach

1. Configuration Driven Approach: The system will use a JSON-based configuration file to define expected arrival times and conditions for different feeds.  
2. Scheduled Execution: A monitoring script will run at regular intervals using an automation scheduler like AutoSys.  
3. Database Query Execution: The script will query the Snowflake database to verify feed arrival.  
4. Alerting Mechanism: If a feed is not found within the expected window, an alert will be triggered to notify relevant teams.  
5. Caching Mechanism: Ensures previously checked and found feeds are not checked again.  
6. Database Tracking: Stores the status of each feed check to allow reporting and avoid redundant checks.

## 2. Configuration Structure

The configuration file contains details about each feed, such as:  
- Feed name  
- Expected start time and end time  
- Frequency of checks  
- Offset for arrival window  
- Snowflake table details  
- Business date and slice time columns  
- Conditions for identifying expected records  
- Alert recipients

## 3. Database Structure

The Snowflake table ARIP.AFRS.CREDIT\_RATING contains:  
- BUS\_DT (Date): The business date for the feed.  
- SLC\_TM (Varchar): The time slice identifier (9999 for End of Day feeds).  
- ID (Number): Unique identifier for each feed record.  
Additionally, a tracking table is required:  
- CHECK\_STATUS: Stores whether the feed was found, missing, or pending.  
- LAST\_CHECK\_TIMESTAMP: Tracks when the last check was performed.

## 4. Execution Flow

1. Load Configuration:  
 - Read configuration from JSON or database table.  
 - Extract feed details.  
  
2. Determine Business Date:  
 - Check if today is a business day using Snowflake database.  
 - If today is a holiday, determine the last valid business day.  
  
3. Process Each Feed:  
 - Loop through each configured feed.  
 - Calculate expected feed arrival time using start\_time and offset values.  
  
4. Check Feed Arrival:  
 - Query Snowflake to check if expected feed record exists.  
 - If feed is found, mark it as 'FOUND' in the tracking table.  
 - If feed is missing, trigger an alert and mark as 'MISSING'.  
  
5. Cache and Track Progress:  
 - Store status in tracking table to avoid redundant checks.  
 - Ensure only pending and missing feeds are checked in subsequent runs.  
  
6. Alerting:  
 - Send alerts to configured recipients if a feed is missing.

## 5. Pseudocode for Monitoring Logic

BEGIN  
 LOAD configuration settings  
 CONNECT to Snowflake database  
   
 DETERMINE business date:  
 QUERY database to check if today is a business day  
 IF holiday, set business date to last valid business day  
  
 FOR EACH feed in configuration:  
 CALCULATE expected check time (start time + offset)  
 QUERY Snowflake for feed arrival status  
   
 IF feed found:  
 UPDATE tracking table: status = 'FOUND'  
 ELSE:  
 UPDATE tracking table: status = 'MISSING'  
 TRIGGER alert  
  
 STORE updated check results  
 CLOSE database connection  
END

## 6. Reporting and Storage

- A dedicated tracking table will store feed status updates.  
- Reports can be generated from this table to analyze feed arrival trends.  
- Real-time monitoring dashboards can be built using this data.

## 7. Example Scenarios

Scenario 1: End of Day Check  
- Configuration: Start time 14:00, frequency -1, offset 3 hours.  
- Execution: Runs at 17:00 to check if feed was available by 14:00.  
- Expected Outcome: If missing, alert is triggered.  
  
Scenario 2: Intraday Hourly Check  
- Configuration: Start time 08:00, frequency { "hours": 1 }, offset { "minutes": 30 }.  
- Execution: Runs every hour, checking for last completed feed.  
- Expected Outcome: If a feed is missing within check window, alert is sent.