SSIS Wireframe Documentation

# 1. Introduction

This document provides a detailed overview of the SSIS package – Package\_Extract. The package is responsible for executing extract definitions, generating files (CSV/Excel), applying transformations, logging activities, handling errors, and sending notifications for both success and failure cases.  
  
The package is designed to handle multiple extract definitions in parallel batches for performance optimization and ensures robust error handling, auditing, and archival of generated files.

# 2. Prerequisites

Before running the package, ensure the following:

• Configuration:  
 - All extract details (such as definition\_id, output\_destination, file\_format, SQL command, archive settings, etc.) must be present in the SSIS\_Config table.  
 - Parameters for extracts (if applicable) should be pre-defined in the parameter table.  
  
• SQL Server Agent Job:  
 - The package should be executed through a scheduled SQL Server Agent Job.  
 - The job ensures that only the scheduled extract definitions run at the configured time.

# 3. Package Overview: Package\_Extract

## 3.1 Components

### Connection Managers

- HHSSVRPTSQLP001.Supplemental.conmgr  
 • Server: HHSSVRPTSQLP001  
 • Database: CURIS

### Parameters

- Package parameters are passed dynamically by the SQL Agent Job.

## 3.2 Tasks Breakdown

Below is a detailed breakdown of each major task in the package:

### Get Extract Definition

Retrieves all scheduled extract definitions for the current job run. Extract details such as definition\_id, destination, file format, and SQL command or file path are collected. The extracts are divided into 5 groups using a mod column to enable parallel processing.

### Foreach Loop Container (per mod batch)

Loops over the extract definitions in the assigned batch. Maps the details retrieved earlier into SSIS variables and parameters that downstream tasks use.

### Insert Audit Log

Inserts a log entry into the audit table marking the process start for each extract definition.

### Execute SQL Task 1

Joins the extract definition with the parameter table to fetch parameters if defined. These parameters are then used during query execution.

### Run and Generate Files

Executes the SQL command to generate the extract file. Two variations exist: one executes SQL directly from the config table, the other from a .sql file path. Timeout settings can be configured via task expressions (-t parameter).

### Handle Query Timeout Errors

Uses a C# script to check for query timeout failures. If successful, the workflow proceeds; if failed, the event handler logs the error and deletes the incomplete file.

### Log Empty Files (Without Header)

C# script checks whether the generated file is empty (ignoring headers). If empty, this information is stored in a variable for later notification.

### Add Text Qualifier

Adds a configurable text qualifier to the generated file. The qualifier setting is driven from the SSIS\_Config table.

### Remove Warning/Hyphen Rows

Cleans the file by removing unwanted messages (e.g., 'rows affected') and extraneous hyphen rows. Also stores the record count into a variable for auditing.

### Log Empty Files (With Header)

Runs if the extract requires a header. Checks the file after data cleaning and determines if it is empty, excluding the header row.

### Convert CSV to Excel

Converts generated CSV files to Excel format. Two variations exist: one handles files with headers, the other without headers.

### Get Final File Size

Captures the size of the final generated file and stores it for reporting and validation.

### Log Success

Writes a success entry into the log table once an extract has been successfully completed.

### Copy File to Archive

If the extract definition is marked with Archive = Yes, the file is copied to the configured Archive\_Location specified in the SSIS\_Config table.

### Get Distinct Recipients (Batch Level)

After batch completion, retrieves all distinct recipients configured to receive notifications for that batch.

### Foreach Loop Container (Recipients)

Iterates through each distinct recipient to process and send notifications.

### Calculate Error Flag

Determines the run status (success or failure) for all definitions assigned to each recipient.

### Send Mail for Errors

Sends an email notification to each recipient that had one or more failed extracts, including extract details and failure reasons.

### Send Batch Status Mail

Sends an email summary of the entire batch to recipients, including overall status and summary of successes/failures.

### Join All Definition IDs

Combines definition IDs from all 5 parallel batches that generated empty files.

### Set Empty File Flag

Updates the isEmptyFlag column to 'Y' for all extract definitions that produced empty files.

### Send Mail for Empty Files

Sends notification emails listing all extracts that generated empty files, including relevant details for recipients.

# 4. Error Handling Strategy

• Timeouts: Captured via C# scripts, logged, and invalid files deleted.  
• Empty Files: Identified post-cleaning, flagged, and included in notification emails.  
• Event Handlers: Automatically trigger on task failure to log errors and ensure cleanup.  
• Email Notifications: Detailed emails sent for failures, empty files, and overall batch status.

# 5. Key Notes

• Parallel Processing: Extracts are split into 5 parallel containers using the mod column.  
• Custom C# Scripts: Widely used for file validation, cleanup, conversion, and transformation.  
• Config Driven: SQL commands, destinations, headers, qualifiers, and archive settings are all driven from SSIS\_Config.