

**Virgin Trains West Coast**

**Bugle Staging Design specification**

Bugle Staging Document

Steve Forster

Technical Consultant

**July 2018**

z

Contents

[1. Document Management 2](#_Toc520123446)

[2. Document Purpose 3](#_Toc520123447)

[2.1 In Document Scope 3](#_Toc520123448)

[2.2 Out of Document Scope 3](#_Toc520123449)

[3. High Level Overview 4](#_Toc520123450)

[3.1. Process Context Diagram 4](#_Toc520123451)

[3.2. Process Flow 5](#_Toc520123452)

[3.3. Mapping 5](#_Toc520123453)

# Document Management

|  |  |  |  |
| --- | --- | --- | --- |
| **Version Control** | | | |
| **Version** | **Date** | **Author(s)** | **Sections Changed** |
| 0.1 | 16/07/2018 | Steve Forster | Initial draft document |

| **Distribution List** | | |
| --- | --- | --- |
| **Organisation** | **Name** | **Role** |
| Merkle | Mark Jones | Client Lead / Project Manager |
| Merkle | Jez Cox | IBM Software SME |
| Merkle | Gary Newsome | Technical Consultant |
| Merkle | John Whittome | Business Analyst |
| Merkle | Peter Malherbe | AWS SME |
| Merkle | Mat Lynd | Development |
| Merkle | Juanjo Diaz | Technical consultant |
| Merkle | Steve Forster | Technical Solution Lead |
| Merkle | Avtar Aswell | Tester |

# Document Purpose

The purpose of this ***Bugle Staging Design Document*** is to define the scope of how the solution will process data for the Tracsis Bugle feed from the pre-processing tables to the Staging tables

The list of functional requirements that will be documented in this design are covered below in the scoping section.

Each pre-processing feed process will be developed as an individual SSiS package to enable scheduling at different times depending on the file arrival details.

The load to staging will be dependent on the pre-processing for each feed to complete without error

This document should be read in conjunction with the Technical Design document and the Business Requirements documents

## 2.1 In Document Scope

The following items are in scope for this document:

* A data flow diagram showing the steps that will be covered as part of the staging process
* Logging of the number of records loaded, rejected and loaded into the processing tables
* The details of the metadata required to support the data feed loads

## 2.2 Out of Document Scope

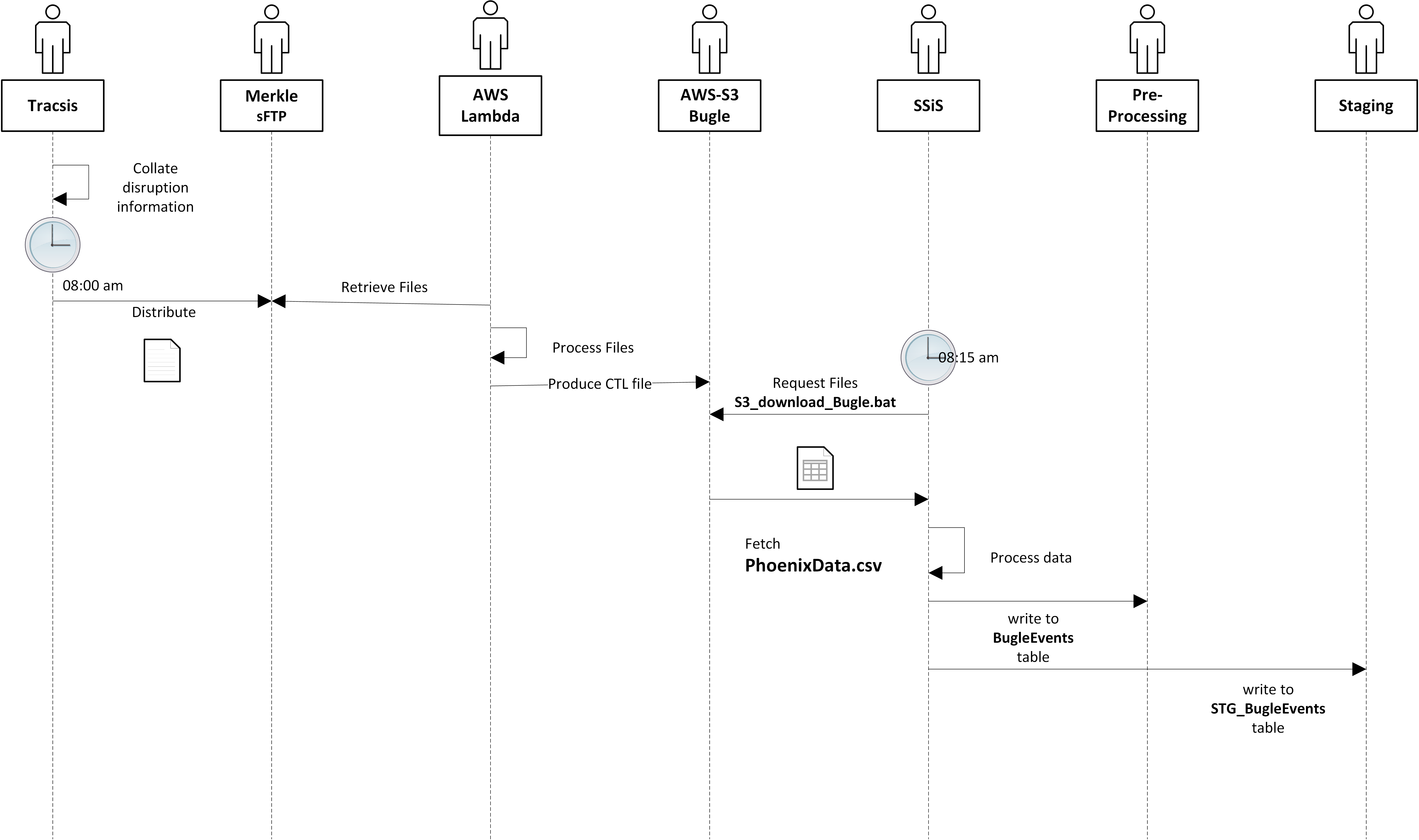
The following items are out of scope for this document and will be covered to separate design documents:

* The SFTP to S3 bucket transfer and logging
  + This is covered as part of the pre-processing load diagram
* The design to process records through to pre-processing

# High Level Overview

## 3.1. Process Context Diagram

Tracsis will produce a daily extract and place on the Merkle sFTP server for retrieval and processing by Virgin Trains West Coast.



The daily file will be loaded to pre-processing, and this document addresses the design of the load to the Staging environment from pre-processing.

## 3.2. Process Flow

The workflow will be invoked and the following logic applied…

## 3.3. Entity Relationship

Start

Records with OperationStatus = ‘Imported’ (4)  
?

Yes

Set record OperationStatus to ‘Processing’ (5)

No

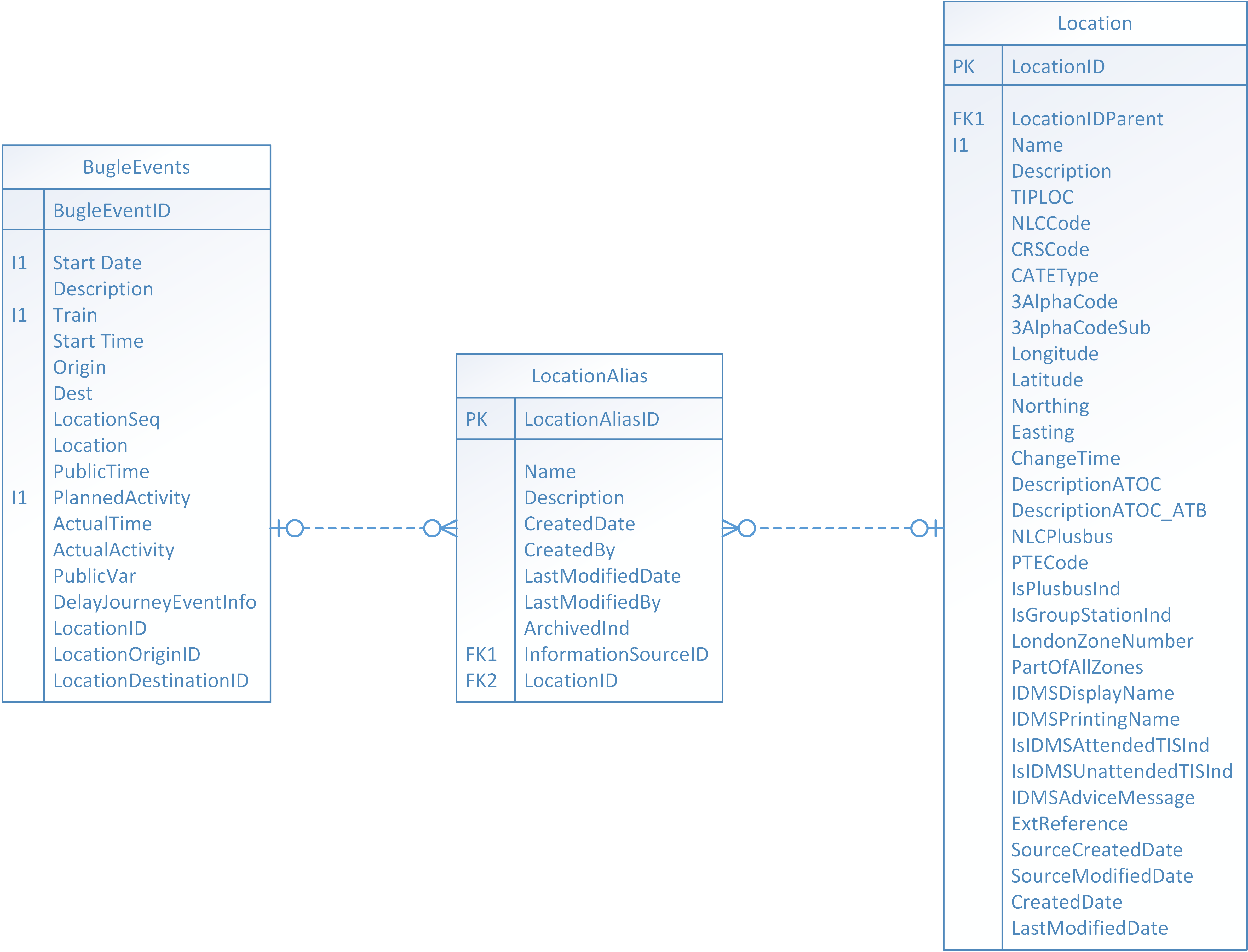
Finish

Insert record into STG\_BugleEvents

Set record OperationStatus to ‘Completed’ (6)

Update Load Log Counts

The [STG\_BugleEvents].[LocationID], [LocationOriginID] and [LocationDestinationID] are Foreign Keys, and are matched using the “Location” from the BugleEvent to “Name” on the LocationAlias table to return the LocationID.



## 3.4. Mapping

STG\_BugleEvents will be populated from the preprocessing.BugleEvents table

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Source Table | Source Column | Source Data type | Primary Key | Transformation  Rule | Comments/ Other Rules | Error Handling Rules | Target Table | Target Column | Target  Data  type |
|  |  |  | Y |  | Identity Column |  | STG\_BugleEvents | BugleEventId | int |
|  |  |  |  |  |  |  | STG\_BugleEvents | CreatedDate | date time |
|  |  |  |  | **0** |  |  | STG\_BugleEvents | ArchiveInd | bit |
| BugleEvents | [Start Date] | varchar |  | convert to ‘yyyy-mm-dd’ | ODBC canonical format |  | STG\_BugleEvents | StartDate | date |
| BugleEvents | [Start Time] | Varchar |  | Convert to ‘hh:mi:ss’ | 24 hour format |  | STG\_BugleEvents | StartTime | time |
| BugleEvents | Train | Varchar |  |  |  |  | STG\_BugleEvents | TrainID | varchar(4) |
| BugleEvents | Description | Varchar |  |  |  |  | STG\_BugleEvents | Description | Varchar(512) |
| BugleEvents | Location | Varchar |  | Lookup FK LocationAlias |  |  | STG\_BugleEvents | LocationID | int |
| BugleEvents | Origin | Varchar |  | Lookup FK LocationAlias |  |  | STG\_BugleEvents | LocationOriginID | int |
| BugleEvents | Dest | Varchar |  | Lookup FK LocationAlias |  |  | STG\_BugleEvents | LocationestinationID | int |
| BugleEvents | LocationSeq | Varchar |  | Convert to Integer |  |  | STG\_BugleEvents | LocationSeq | Int |
| BugleEvents | PlannedActivity | Varchar |  | Process record if  planned activity <> ‘P’ | Valid values - A D O T |  | STG\_BugleEvents | PlannedActivity | char(1) |
| BugleEvents | PublicTime | Varchar |  |  |  |  | STG\_BugleEvents | PublicTime | time |
| BugleEvents | ActualTime | Varchar |  |  |  |  | STG\_BugleEvents | ActualTime | time |
| BugleEvents | ActualActivity | Varchar |  |  | Valid Values -ARRIVE CALV CAPE DEPART DVTD FTS  N/R  ORIGIN PINE TERMINUS |  | STG\_BugleEvents | ActualActivity | varchar(512) |
| BugleEvents | PublicVar | Varchar |  |  |  |  | STG\_BugleEvents | PublicVar | int |
| BugleEvents | DelayJurneyEventInfo | varchar |  |  |  |  | STG\_BugleEvents | DelayJourneyEventInfo | Varchar(512) |

**END**