

**Virgin Trains West Coast**

**Timetable Load Design specification**

Rail Timetable Load Document

Steve Forster

Technical Consultant

**July 2018**

z

Table of Contents

[1. Document Management 2](#_Toc520455870)

[2. Document Purpose 3](#_Toc520455871)

[2.1 In Document Scope 3](#_Toc520455872)

[2.2 Out of Document Scope 3](#_Toc520455873)

[3. High Level Overview 4](#_Toc520455874)

[3.1. Process Context Diagram 4](#_Toc520455875)

[3.2. Process Flow 5](#_Toc520455876)

[3.4. National Rail Data 7](#_Toc520455877)

[3.5 Journey XSLT 9](#_Toc520455878)

[3.6. Journey Stages XSLT 10](#_Toc520455879)

[3.7. Mapping 11](#_Toc520455880)

# Document Management

|  |  |  |  |
| --- | --- | --- | --- |
| **Version Control** | | | |
| **Version** | **Date** | **Author(s)** | **Sections Changed** |
| 0.1 | 27/07/2018 | Steve Forster | Initial draft document |
| 0.2 | 10/10/18 | Steve Forster | cancelReason is from Journey data |

| **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 ***Rail Timetable Staging Design Document*** is to define the scope of how the solution will process data for the National Rail timetable feed from the pre-processing tables through 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.

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

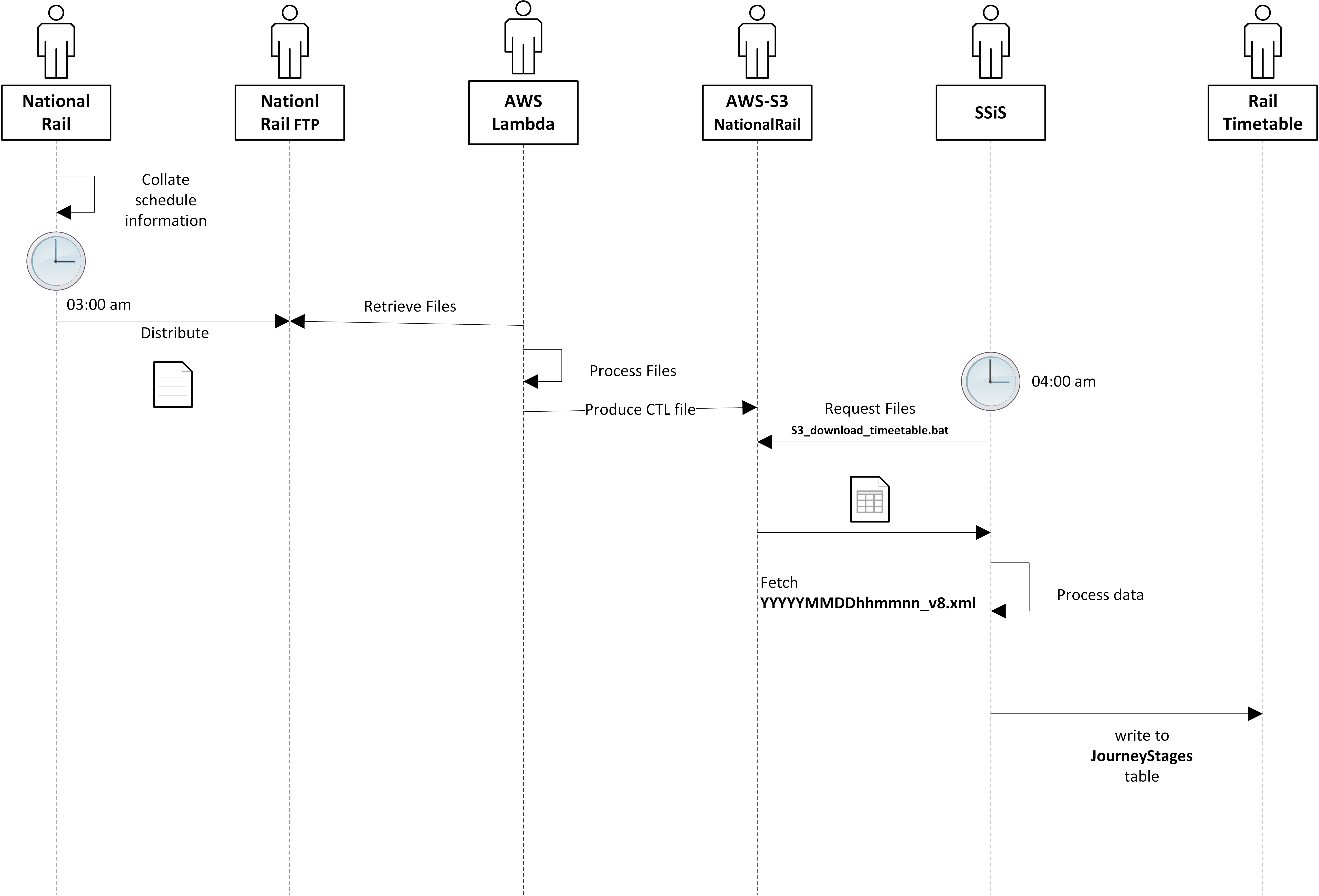
The following items are out of scope for this document and will be covered to separate design documents:

* 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

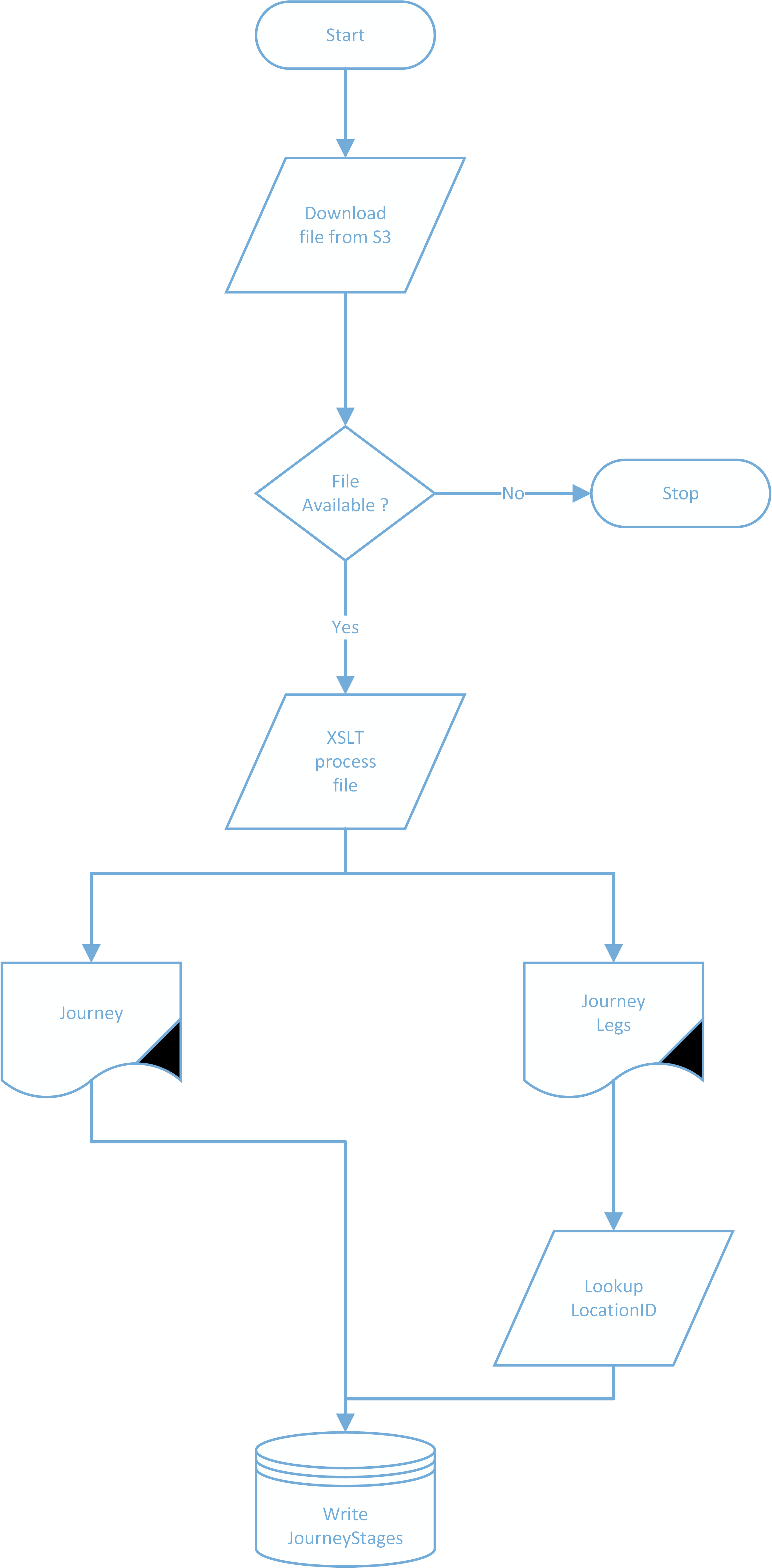
National Rail will produce a daily extract and place on the **datafeeds.nationalrail.com** FTP server for retrieval and processing by Virgin Trains West Coast.



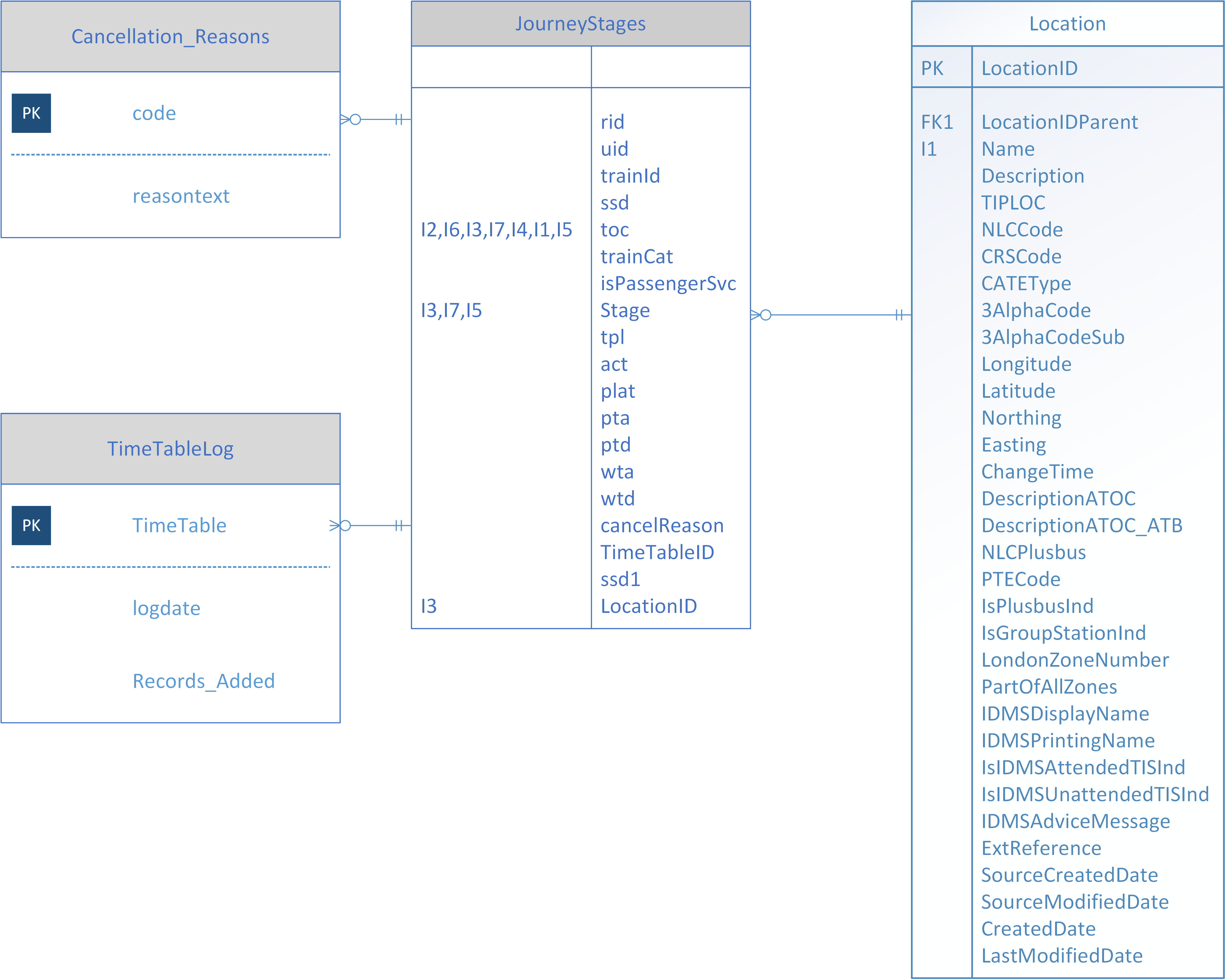
The daily file will be parsed using XSLT processes and written to the JourneyStages table.

## 3.2. Process Flow

The workflow will be invoked, and the following logic applied…

3.3. Entity Relationship

The [JourneyStages].[LocationID is a Foreign Key to [Reference].[Location], and matched using the Timing Point Location [tpl] column from the JourneyStages table to [TIPLOC] on the Location table to return the LocationID.



## 3.4. National Rail Data

National Rail provide a daily file in XML format containing all train scheduled movements.

Journey has the following attributes

|  |  |  |
| --- | --- | --- |
| Column | Description | |
| rid | RTTI unique Train ID | |
| uid | Train UID | |
| trainId | Train ID (headcode) | |
| ssd | Scheduled Start Date | |
| toc | ATOC code | |
| trainCat | Category of Service | |
| cancelReason | Reason for cancellation of service/location | |
| isPassengerSvc | True if Darwin classifies the train category as a passenger service | |
| OR | Passenger Origin calling point | |
|  | **wta** | Working scheduled Time of Arrival |
| **wtd** | Working scheduled Time of Departure |
| **plat** | Platform Number (optional) |
| **pta** | Public Scheduled Time of Arrival |
| **ptd** | Public Scheduled Time of Departure |
| IP | Passenger Intermediate calling point | |
|  | **wta** | Working scheduled Time of Arrival |
| **wtd** | Working scheduled Time of Departure |
| **plat** | Platform Number (optional) |
| **pta** | Public Scheduled Time of Arrival |
| **ptd** | Public Scheduled Time of Departure |
| PP | Passing Point – used for timing purposes | |
|  | **wta** | Working scheduled Time of Arrival |
| **wtd** | Working scheduled Time of Departure |
| **plat** | Platform Number (optional) |
| **pta** | Public Scheduled Time of Arrival |
| **ptd** | Public Scheduled Time of Departure |
| DT | Passenger Destination calling point | |
|  | **wta** | Working scheduled Time of Arrival |
| **wtd** | Working scheduled Time of Departure |
| **plat** | Platform Number (optional) |
| **pta** | Public Scheduled Time of Arrival |
| **ptd** | Public Scheduled Time of Departure |

We are only interested in loading Passenger journeys pertaining to Virgin Trains West Coast, and filter on TOC=’VT’ and ignore the ‘PP’ records.

Example XML

The snippet below shows the structure of the XML received and shows a single journey on a VTWC route from Wolverhampton to London Euston, for a train with a RSID of **1B46** departing 27-June 22:45, arriving at Euston 01:13.

<?xml version="1.0" encoding="utf-8"?>

<PportTimetable xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" timetableID="20180627020728" xmlns="http://www.thalesgroup.com/rtti/XmlTimetable/v8">

<Journey rid="201806276770352" uid="C70352" trainId="1B46" ssd="2018-06-27" toc="VT" trainCat="XX">

<OR tpl="WVRMPTN" act="TB" plat="4" ptd="22:45" wtd="22:46" />

<PP tpl="DUDLPT" wtp="22:53" />

<IP tpl="SNDWDUD" act="T " plat="1" pta="22:54" ptd="22:55" wta="22:55" wtd="22:56:30" />

<PP tpl="GALTONJ" wtp="22:59:30" />

<PP tpl="SOHOSJ" wtp="23:03" />

<IP tpl="BHAMNWS" act="T " plat="4" pta="23:06" ptd="23:10" wta="23:07" wtd="23:10" />

<PP tpl="PROOFHJ" wtp="23:12:30" />

<PP tpl="STECHFD" wtp="23:15" />

<IP tpl="BHAMINT" act="T " plat="4" pta="23:19" ptd="23:20" wta="23:18:30" wtd="23:20:30" />

<IP tpl="COVNTRY" act="T " plat="1" pta="23:30" ptd="23:31" wta="23:30" wtd="23:32" />

<PP tpl="RUGBTVJ" wtp="23:41:30" />

<IP tpl="RUGBY" act="T " plat="5" pta="23:43" ptd="23:44" wta="23:42:30" wtd="23:44:30" />

<PP tpl="HMTNJ" wtp="23:47:30" />

<PP tpl="DVNTYNJ" wtp="23:49:30" />

<PP tpl="LNGBKBY" wtp="23:54" />

<IP tpl="NMPTN" act="D " plat="1" pta="00:05" wta="00:05" wtd="00:07" />

<PP tpl="HANSLPJ" wtp="00:16" />

<IP tpl="MKNSCEN" act="T " plat="1" pta="00:23" ptd="00:24" wta="00:22:30" wtd="00:24" />

<PP tpl="BLTCHLY" plat="4" wtp="00:27" />

<PP tpl="LEDBRNJ" wtp="00:37:30" />

<PP tpl="TRING" plat="5" wtp="00:41:30" />

<PP tpl="BONENDJ" wtp="00:45:30" />

<IP tpl="WATFDJ" act="D " plat="9" pta="00:52" wta="00:52" wtd="00:53:30" />

<PP tpl="HROW" plat="6" wtp="00:58" />

<PP tpl="WMBY" plat="6" wtp="01:02" />

<PP tpl="WLSDWLJ" wtp="01:07" />

<PP tpl="CMDNJN" wtp="01:10" />

<PP tpl="CMDNSTH" wtp="01:10:30" />

<DT tpl="EUSTON" act="TF" pta="01:13" wta="01:13" />

</Journey>

</PportTimetable>

## 3.5 Journey XSLT

Because SSiS XML import does not support XML schemas with complex types we use XSLT to transform the XML into a CSV file with a row per train journey so that we can ingest the data easier.

A filter is applied to only load journeys for Virgin Trains West Coast, and each row is terminated with a line feed.

* Note SSiS does not support the hexadecimal representation **0x0A;**   
  so we use decimal **&#10;** for the linefeed character.

<?xml version="1.0" encoding="UTF-8"?>

<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform">

<xsl:output method="text"/>

<xsl:strip-space elements="\*"/>

<xsl:variable name="delimiter" select="','"/>

<xsl:template match="/\*">

<xsl:variable name="timetableID" select="@timetableID"/>

<xsl:for-each select="\*">

<xsl:variable name="rid" select="@rid"/>

<xsl:variable name="uid" select="@uid"/>

<xsl:variable name="trainId" select="@trainId"/>

<xsl:variable name="ssd" select="@ssd"/>

<xsl:variable name="toc" select="@toc"/>

<xsl:variable name="trainCat" select="@trainCat"/>

<xsl:variable name="isPassengerSvc" select="@isPassengerSvc"/>

<xsl:variable name="cancelReason" select="@cancelReason"/>

<xsl:if test="name() = 'Journey' and @toc='VT' ">

<xsl:value-of select="$timetableID"/> <xsl:value-of select="$delimiter"/>

<xsl:value-of select="$rid"/> <xsl:value-of select="$delimiter"/>

<xsl:value-of select="$uid"/> <xsl:value-of select="$delimiter"/>

<xsl:value-of select="$trainId"/> <xsl:value-of select="$delimiter"/>

<xsl:value-of select="$ssd"/> <xsl:value-of select="$delimiter"/>

<xsl:value-of select="$toc"/> <xsl:value-of select="$delimiter"/>

<xsl:value-of select="$trainCat"/> <xsl:value-of select="$delimiter"/>

<xsl:value-of select="$isPassengerSvc"/> <xsl:value-of select="$delimiter"/>

<xsl:for-each select="\*">

<xsl:if test="name() = 'cancelReason' ">

<xsl:value-of select="."/>

</xsl:if>

</xsl:for-each>

<xsl:text>&#10;</xsl:text>

</xsl:if>

</xsl:for-each>

</xsl:template>

</xsl:stylesheet>

## 3.6. Journey Stages XSLT

Because SSiS XML import does not support XSD with complex types we use XSLT to transform the XML into a CSV file with a row per train stop so that we can ingest easier.

A filter is applied to only load the Origin (OR), Destination (DT) and Scheduled (IP) stops.

Each row is terminated with a line feed.

<?xml version="1.0" encoding="UTF-8"?>

<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform">

<xsl:output method="text"/>

<xsl:strip-space elements="\*"/>

<xsl:preserve-space elements="Journey Association" />

<xsl:variable name="delimiter" select="','"/>

<xsl:variable name="tpl" select="'aaa'"/>

<xsl:variable name="elements\_list" select="'OR IP DT'" />

<xsl:template match="/\*/\*">

<xsl:variable name="rid" select="@rid"/>

<xsl:variable name="uid" select="@uid"/>

<xsl:variable name="toc" select="@toc"/>

<xsl:for-each select="\*">

<xsl:if test="contains(

concat(' ', $elements\_list, ' '),

concat(' ', name(), ' ')

)">

<xsl:value-of select="$rid"/> <xsl:value-of select="$delimiter"/>

<xsl:value-of select="$uid"/> <xsl:value-of select="$delimiter"/>

<xsl:value-of select="name()"/> <xsl:value-of select="$delimiter"/>

<xsl:value-of select="@tpl"/> <xsl:value-of select="$delimiter"/>

<xsl:value-of select="@act"/> <xsl:value-of select="$delimiter"/>

<xsl:value-of select="@plat"/> <xsl:value-of select="$delimiter"/>

<xsl:value-of select="@pta"/> <xsl:value-of select="$delimiter"/>

<xsl:value-of select="@ptd"/> <xsl:value-of select="$delimiter"/>

<xsl:value-of select="@wta"/> <xsl:value-of select="$delimiter"/>

<xsl:value-of select="@wtd"/> <xsl:text>&#10;</xsl:text>

</xsl:if>

</xsl:for-each>

</xsl:template>

</xsl:stylesheet>

## 3.7. Mapping

[RailTimetable].[JourneyStages] will be populated from the Journey and Journey Stages CSV files

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Source | Source Column | Source Data type | Primary Key | Transformation  Rule | Comments/ Other Rules | Error Handling Rules | Target Table | Target Column | Target  Data  type |
| Journey | rid | varchar | Y |  |  |  | JourneyStages | rid | varchar(512) |
| Journey | uid | varchar |  |  |  |  | JourneyStages | uid | varchar(512) |
| Journey | trainid | varchar |  |  |  |  | JourneyStages | trainid | varchar(512) |
| Journey | ssd | varchar | Y |  | Date in YYYY-MM-DD format |  | JourneyStages | ssd | date |
| Journey | toc | varchar |  |  |  |  | JourneyStages | toc | varchar(8) |
| Journey | trainCat | varchar |  |  |  |  | JourneyStages | trainCat | varchar(4) |
| Journey | isPassengerSvc | varchar |  |  |  |  | JourneyStages | isPassengerSvc | bit |
| Journey | cancelReason | varchar |  |  |  |  | JourneyStages | cancelReason | varchar(512) |
| Journey Stages | stage | varchar | Y |  |  |  | JourneyStages | stage | varchar(2) |
| Journey Stages | tiploc | varchar |  |  |  |  | JourneyStages | tpl | varchar(8) |
| Journey Stages | activity | varchar |  |  |  |  | JourneyStages | act | varchar(4) |
| Journey Stages | platform | varchar |  |  |  |  | JourneyStages | plat | varchar(8) |
| Journey Stages | pta | varchar |  |  |  |  | JourneyStages | pta | time |
| Journey Stages | ptd | varchar |  |  |  |  | JourneyStages | ptd | time |
| Journey Stages | wta | varchar |  |  |  |  | JourneyStages | wta | time |
| Journey Stages | wtd | varchar |  |  |  |  | JourneyStages | wtd | time |
| Journey Stages | LocationID | varchar |  |  | Lookup tiploc on Reference.Location table |  | JourneyStages | LocationID | int |
| Journey Stages | timetableID | varchar |  |  |  |  | JourneyStages | timetableID | varchar(20) |

**END**