Talend Data Quality Framework – Architecture and Prerequisites Guide

Introduction

This document provides an overview of the Talend Data Quality Framework (DQF) architecture and details of the infrastructure prerequisites needed to install and operate the framework.

Prerequisites

Subscription requirements

A current subscription to one or both of the following Talend Accelerator Services:

- Enable Analytics
- Establish Data Excellence

Contents

Solution architecture

Key framework features

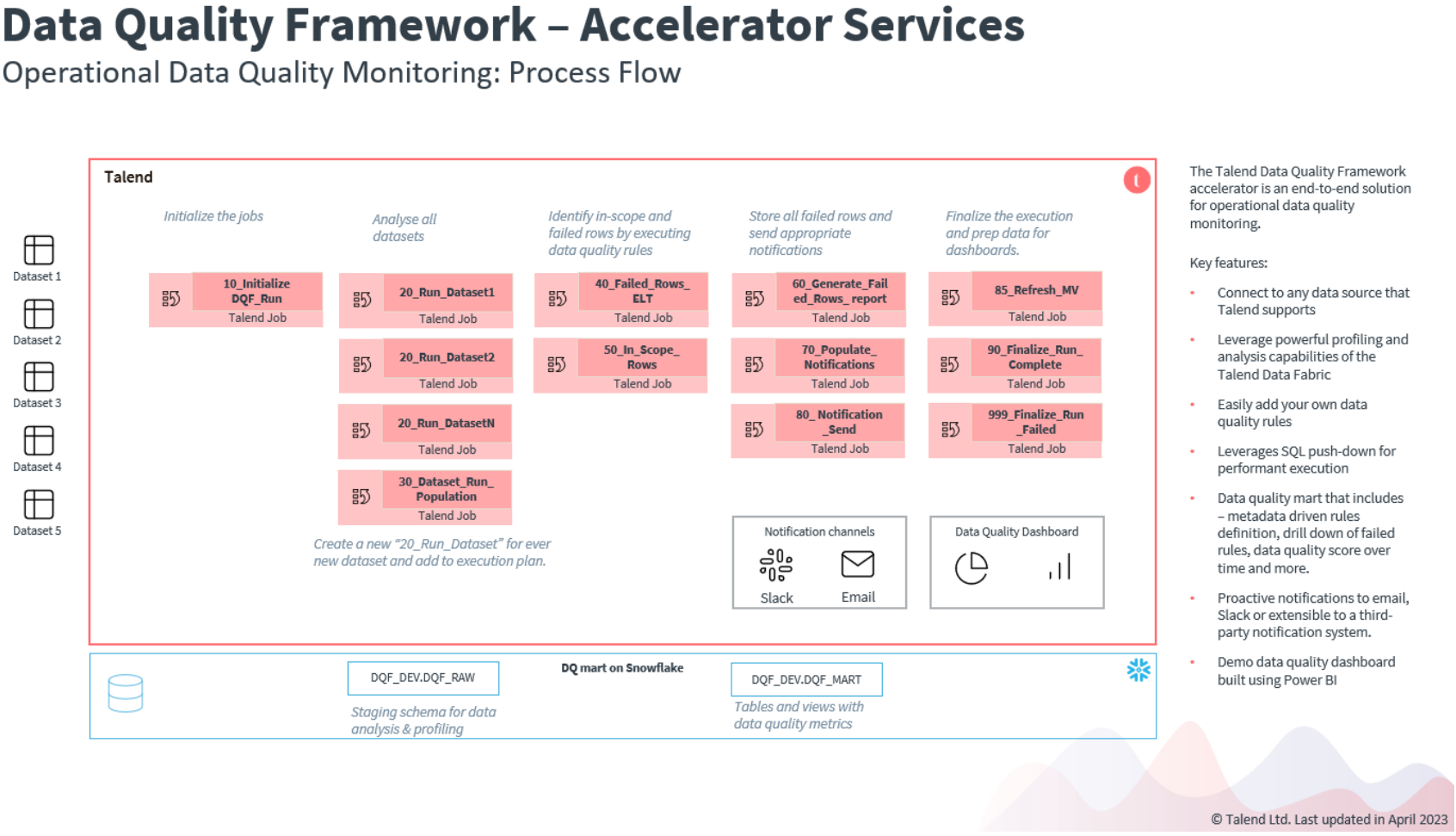
- Connects to nearly any data source
- Leverages the powerful analysis capabilities of Talend Data Fabric to help analyze your datasets and define rules
- Includes a wide range of analysis indicators and data quality rules
- Allows you to easily add your own data quality rules– from simple to highly complex
- Leverages SQL pushdown for performance execution
- Aligns with the DAMA (UK) data quality dimensions:
    - Accuracy
    - Completeness
    - Timeliness
    - Uniqueness
    - Consistency
    - Validity
- Uses a data mart that provides the following features:
    - Metadata-driven rules definition
    - Failed rules drilldown
    - Data quality scoring over time
    - Metadata-driven notifications
    - Easily consumable reports
- Sends proactive notifications to email, Slack, or a third-party notification system of your choice (for example, Data Stewards get notifications when data quality issues arise)
- Contains a demo dashboard built using Power BI– demonstrates how to build a dashboard on top of the mart in the Business Intelligence tool of your choice

Services

To enable you to get maximum value from Talend DQF, Talend provides a range of Value Added Services that are focused on the framework as part of your Accelerator Services Subscription. By leveraging these services, you can achieve maximum time to value.

Process view

The diagram below provides an overview of the individual steps that execute during a Talend DQF run.



The Talend Data Quality Framework accelerator is an end-to-end solution for operational data quality monitoring.

Key features:

- Connect to any data source that Talend supports
- Leverage powerful profiling and analysis capabilities of the Talend Data Fabric
- Easily add your own data quality rules
- Leverages SQL push-down for performant execution
- Data quality mart that includes – metadata driven rules definition, drill down of failed rules, data quality score over time and more.
- Proactive notifications to email, Slack or extensible to a third-party notification system.
- Demo data quality dashboard built using Power BI

Talend provides the majority of steps in this process as pre-compiled binaries. These binaries are as follows:

- DQF_10_Initialise_DQF_Run
- DQF_30_Dataset_Run_Population
- DQF_40_Failed_Rows_ELT
- DQF_50_In_Scope_Rows_ELT
- DQF_60_Generate_Failed_Rows_Report
- DQF_70_PopulateNotifications
- DQF_85_Refresh_Materialized_Views
- DQF_90_Finalize_Run_Complete

The installation package provides a script that deploys these binaries to your Talend Cloud tenant.

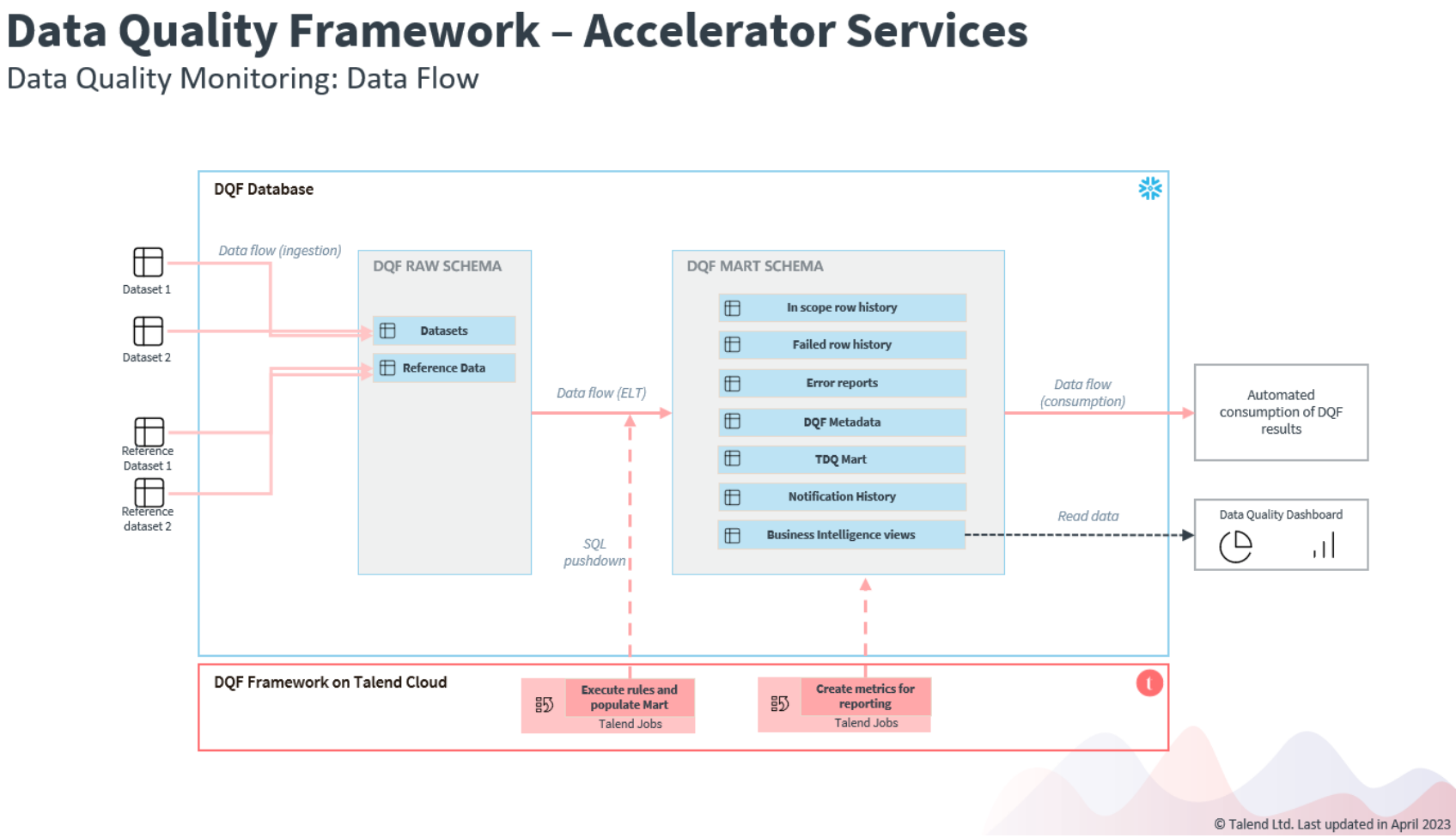
The Jobs may execute on a Talend cloud engine or Remote Engine, with the following considerations:

- The Studio version and monthly release requirements must be strictly met
- If running on a cloud engine, the database containing the DQF schemas must be accessible by the cloud engine

Talend DQF Developer users can develop the Jobs not set up as binaries and deploy them to Talend Cloud to execute as part of the overall process flow. The framework also bundles additional example Jobs for data ingestion, hooks for extensibility, and numerous interfaces for results consumption.

Data flow view

Talend Data Quality Framework is designed to be flexible, but performant at scale. For this reason, the v1 release of the framework primarily leverages the ELT (Extract Load Transform, otherwise known as SQL pushdown) approach to execution. By doing so, the framework is able to leverage the computing power of modern cloud database solutions without needing to pull large volumes of data out of the database and into Talend Data Integration Jobs. The data flow diagram below illustrates this process:



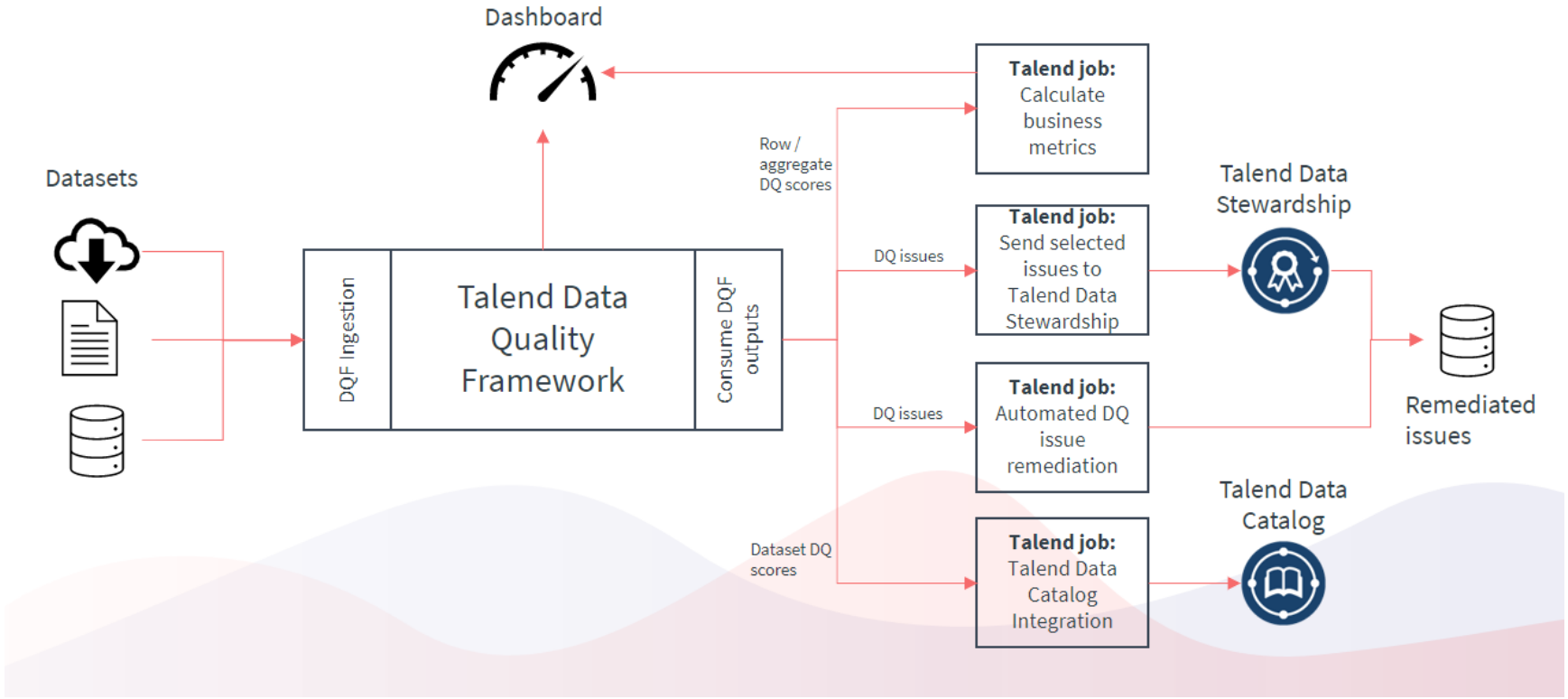
By operating in this mode, a snapshot (copy) of each dataset under management by the framework is taken at the beginning of each execution run. This ensures that the data is not changing (for example, by new transactions) during data analysis and data quality rules execution. Other Talend DQF modes of operation are planned for future releases.

A DQF run can contain 1.n datasets and framework users can schedule different runs of the framework, for example at different times or with different datasets. The only stipulation in v1 is that two instances of a DQF Run cannot overlap– one run must finish before the next run commences.

Note: The diagram above shows the data flow for in-production datasets under management by the framework. The onboarding process, as documented in the **DQF dataset onboarding guide**, has the option to leverage Talend Cloud Data Inventory to aid in semantic analysis of the dataset, which entails data flowing to Talend Cloud. An alternative approach for those wishing to avoid the flow of sensitive data to Talend Cloud is to leverage a hybrid deployment of Talend Data Preparation. You can familiarize yourself with the pros and cons of this approach by reviewing the Talend Cloud Hybrid Installation Guide for Linux.

DQF as part of a wider solution architecture

It is the expectation that the Talend Data Quality Framework will be a component of a wider solution architecture for many customers. The framework provides pre-defined interfaces for ingestion of data for analysis and consumption of DQ Framework results. An example of a wider solution architecture is shown below.



It is important to note that the framework itself has limited and pre-defined points of extensibility. For example, customers can add support for their own notification channels (email and Slack support is provided out of the box). Changes to the core Framework itself must be requested as a feature request, which DQF customers are able to submit to Talend.

Prerequisites

Talend Cloud environment

Talend recommends that you review the guidance provided in the Talend Cloud Physical Reference Architecture prior to standing up your Talend Cloud environment. The steps required to deploy Talend Cloud are out of scope for this document, however there are the following infrastructure requirements:

- Talend Studio 8 (for monthly release requirements, see the engines below)
- A Git repository and associated Talend project for the framework Jobs
- An artifact Repository (used as part of the standard Talend architecture for developer library sharing and patch management)
- A supported SQL database and appropriate SQL client software
- A Talend Engine, Cloud or Remote:
    - Cloud Engine:
        - Cloud engines use Java 8 (May 2023), therefore the developer studio must be patched to **R2023-04** at minimum
    - Remote Engine:
        - If using Java 8, the developer studio must be patched to **R2023-04** at minimum
        - If using Java 11, the developer studio must be patched to **R2022-12** at minimum
- Talend Remote Engine Generation Two (optional, but recommended)
- Power BI desktop (for the demo dashboard supplied with the framework)
- A business intelligence tool of your choice

Further details are provided in subsequent sections of this document.

Git repository or Talend project

Talend recommends that you use a single Talend project per Git repository. It is also recommended that you store DQF artifacts (both those provided by Talend and those developed by you) in their own Git repository or Talend project, separate from other projects and artifacts (however this is not mandatory).

Database

For release v1.1.0, you must use Snowflake or MySQL 8 (or a compatible cloud provider, for example, AWS Aurora). Future releases will add support for other databases. However, as the framework leverages the Talend Data Quality reporting mart, the supported list of DQF databases will always be a subset of those supported by Talend (for a list, see the Supported databases for the data mart page).

Snowflake

The framework requires two schemas in the same Snowflake database, both accessible by the same user. For installation purposes, you should grant full object rights for these two schemas to the user performing the installation. The **DQF installation guide** recommends the following naming conventions:

- **DQF Database**: DQF
- **DQF Analysis Schema**: DQF_RAW
- **DQF Mart Schema**: DQF_MART

You must also have a Snowflake warehouse and a Snowflake role. If you choose to use the supplied Power BI demo dashboard, you may wish to create separate, read-only users to access the mart.

MySQL

The framework requires two schemas in the same MySQL database, both accessible by the same user. For installation purposes, you should grant full object rights for these two schemas to the user performing the installation. The **DQF installation guide** recommends the following naming conventions:

- **DQF Analysis Schema**: DQF_RAW
- **DQF Mart Schema**: DQF_MART

If you choose to use the supplied Power BI demo dashboard, you may wish to create separate, read-only users to access the mart.

The following MySQL privileges are required to install and operate the framework:

- Select
- Insert
- Update
- Create
- Drop
- Index
- Alter
- Create View
- Show View
- References
- Delete

Talend Remote Engines

Remote Engine Generation One

You need a Talend Cloud Engine, Talend Remote Engine Generation One (REG1), or a cluster of Remote Engines to execute the main operational processes of the framework. The framework has the following Remote Engine requirements:

- Ability to communicate with the DQF database (using firewall rules)
- Ability to communicate with any required source systems (for data ingestion)
- Ability to communicate with desired notification systems, such as Slack or email
- Sufficient memory capacity to execute the DQF process at its scheduled execution times. As DQF primarily operates in SQL pushdown mode in this release, the compute, memory, disk, and network requirements of the framework are minimal. Jobs do not usually consume more than their configured maximum amount of memory (1 gigabyte), and most will consume much less than this.

Remote Engine Generation Two

A Talend Remote Engine Generation Two (REG2) must support the following processes:

- Semantic analysis of a dataset located behind your firewall and inaccessible by Talend Cloud during the onboarding process
- Data ingestion using Talend Pipeline Designer (see **DQF dataset onboarding guide**)

Alternatively, you can also use a hybrid deployment of Talend Data Preparation for semantic analysis. There is no hybrid alternative for ingestion using Pipeline Designer.

Power BI

Talend provides an example Power BI dashboard with the framework. You use this dashboard as part of the installation test process. To use this desktop, download and install a PowerBI desktop from the Microsoft website.

Notifications

Proactive notifications of data quality issues are a key feature of the framework. This release delivers out-of-the-box support for email and Slack.

- For email support, you must have an SMTP server and appropriate user.
- For Slack support, you must create a Slack application. To learn more, see the Send Slack messages with Talend article on the Talend website.

Talend Cloud user(s)

One or more Talend Cloud users must fulfill the following DQF personas. Talend Cloud pre-defined user roles for each are as follow:

DQ Developer
Studio – Developer
Operator
Sets appropriate workspace permissions

Business User
Dataset Manager (for license with advanced inventory)
Data Preparation Manager

Application/Data Owner
Data Preparation Administrator
Connection Manager
Dataset Administrator

TMC Operations
Operations – Manage

The above list of roles and permissions is non-exhaustive, and is meant to be tailored to your specific needs.

Multiple DQF environments

This Talend DQF release is set up to support a single environment (production). It is usually desirable to analyze production data (not mockup or test data) when attempting to define data quality rules, so having a single DQF environment is the simplest option. There is no technical restriction that prevents the use of multiple environments (for example, Development, Test, Pre-Production); however, this release does not include any artifacts or processes to automate the promotion of DQF metadata from one environment to another. Future releases may provide this automation and a promotion process compatible with Talend's continuous integration capabilities.

Required information

Use the table below to ensure you have all the pre-requisites necessary to begin DQF installation.

Component	Value	Comments
All server components	OS: Java version:	The Java version must be the same for all installed servers
User (Studio) Client VM	Host: User credentials: VPN details: Studio installed? Studio patch level (see above for requirements): SQL client installed? Network/firewall connectivity to DQF database? Network/firewall connectivity to source systems? Network/firewall connectivity to artifact repository?	Specify any details required to connect. Alternatively provide credentials to Talend using an agreed upon mechanism
Batch (REG1) server	Host:	

Component	Value	Comments
Inventory Pipeline Preparation (REG2) server	Docker version: User credentials:	Alternatively provide data prep hybrid details (if applicable)
Git service	Version: URL: User credentials (if applicable): DQF repository:	Alternatively provide credentials to Talend using an agreed upon mechanism
RDBMS	Type/Version: Jdbc Url: User credentials: Database: DQF raw schema: DQF mart schema: Warehouse: Role:	User should be a service account that will be used to execute the framework on an ongoing basis. Alternatively provide credentials to Talend using an agreed upon mechanism
Monitoring: Email	Host: Port: User credentials: SSL details:	Alternatively provide credentials to Talend using an agreed upon mechanism