TAQ01 V1.0.0 - API Specification

1.Document Change Control

Project Code: TAQ01

Interface Type: File Transfer

Interface Name: ReceiveTaricFile

Author: Vish Murugesan

Date: 21.08.2018

MDG Specification Version: V1.0.0

MDG Specification Status: Draft

SDD: Tariff and Quota Solution Design Document v2.0.pdf

SDD Crossreference/Feature: DIT file integration (SDD v2.0 data flow 1, 2 and 3)

1.1 Version Control

Version	Change	Updated by	Updated Date
V0.1	Initial Creation	Vish Murugesan	19.03.2018
V0.2	1. Added sample metadata file - section 14 2. Updated DIT folder structure - section 2 3. Updated file transfer scenarios - section 9 4. Added note to risk 1 - section 8.1	Vish Murugesan	09.04.2018
V0.3	1. Added scope item 5, out of scope item 3 and dependencies 2 & 3. 2. Added item 9 in section 9. 3. Updated metadata filename example under section 5	Vish Murugesan	26.04.2018
V0.4	1. Updated naming convention 2. Added section 8.5	Vish Murugesan	02.05.2018
V0.5	Added scenario assumptions	Vish Murugesan	20.05.2018
V0.6	Added multiple source file names	Vish Murugesan	17.06.2018
V0.7	Added details to cover the SOAP call to Tariff	Vish Murugesan	16.07.2018
V1.0.0	Added SOAP call to Quota Allocation Module	Vish Murugesan	21.08.2018

2. Transfer Service Properties

Property	Value
Source System	DIT
Soure System Type	FTP Server
Source System OS	Ubuntu
MDG Interface Name	ReceiveTaricFile
Interface Version	V1
Source Compression	None
Source Encoding	UTF-8

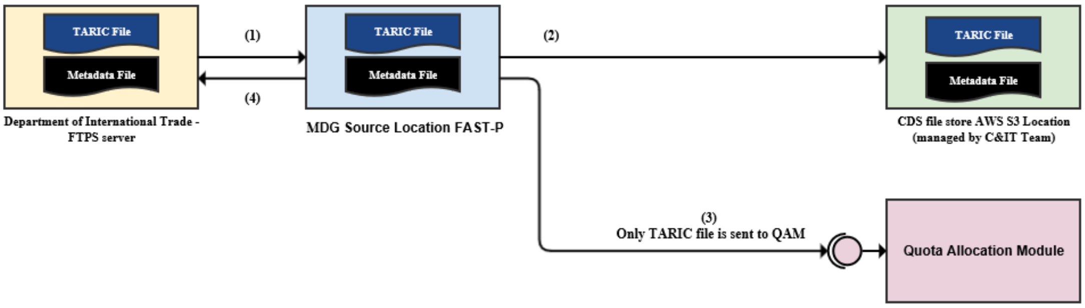
Source location	DIT FTP Server **Folder structure** **tohmrc** – DIT will place the TARIC file in this folder **tohmrc/success** – HMRC will move the TARIC file to this folder if the file was successfully transferred to ED-Tariff **tohmrc/failure** – HMRC will move the TARIC file to this folder in error scenarios **tohmrc/staging** - HMRC will move the TARIC file to this folder while processing the file
Source Filename	*DIT_<Start Date - YYYYMMDD>-<End Date - YYYYMMDD>-<Timestamp - YYYYMMDDTHHMMSS>-EUFileSequence.XML* *DIT_<Start Date - YYYYMMDD>-<End Date - YYYYMMDD>-<Timestamp - YYYYMMDDTHHMMSS>-EUMODFileSequence.XML* *DIT_<Start Date - YYYYMMDD>-<End Date - YYYYMMDD>-<Timestamp - YYYYMMDDTHHMMSS>-NATIONALFileSequence.XML*
Source File Type	XML
Source File Frequency	Daily (file will not be supplied to HMRC during weekends and UK bank holidays)
Expected Source File Size	**Project team to clarify**
Destination System	1. CDS file store (S3 bucket) 2. Quota allocation module
Destination Location	1. CDS File store To QAM folder **(to be confirmed by the project team)** **Folder Structure** ToQAM Success Failure 2. SOAP call to QAM
Destination FileName	File name will remain unchanged from the file name provided by DIT
Destination File Type	Base 64 encoded
Destination Encoding	BASE64
Destination Compression	Zip
Maximum agreed file size	***Up to 50 MB (up to 100 MB post encoding) - Project team to clarify***

3.Objective

The objective of this interface is to transfer the TARIC file from DIT to ED-TARIFF. Following are the high-level steps that the interface will perform:

1. Pick the TARIC file and its corresponding metadata file from DIT FTP Server
2. Move TARIC file and the metadata file from MDG source location to CDS file store S3 location
3. Zip and Encode the file, POST the TARIC file as a payload in a SOAP request to QAM.
4. Handle exception scenarios detailed in this document

   The below block diagram depicts the steps at a high-level.

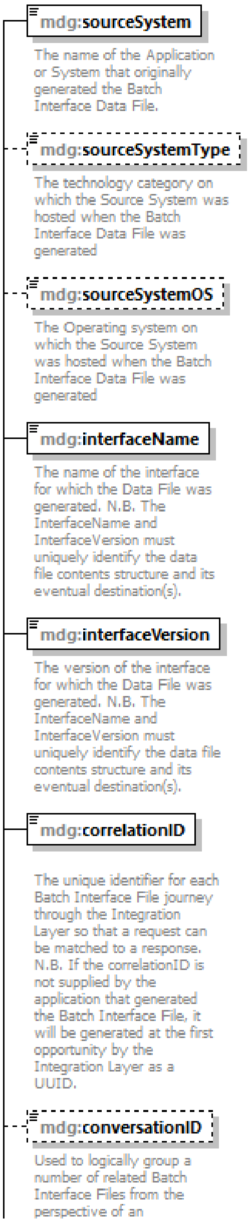

TARIC File
Metadata File
Department of International Trade - FTPS server
(1)
(4)
TARIC File
Metadata File
MDG Source Location FAST-P
(2)
TARIC File
Metadata File
CDS file store AWS S3 Location (managed by C&IT Team)
(3) Only TARIC file is sent to QAM
Quota Allocation Module

4. Metadata File Generation

1. **DIT must generate a valid metadata file for every TARIC file that it produces.**
2. MDG will **reject** the TARIC file in the following scenarios:
   - if the corresponding metadata file does not exist
   - if the checksum value provided in the metadata file does not match the actual checksum of the TARIC file
   - if the metadata file does not confirm to MDG schema
   - if all the mandatory elements are not provided in the metadata file

The TARIC XML file generated by DIT must confirm to the below schema :

file: BatchFileInterfaceMetadata-1.0.7.xsd



mdg:sourceSystem

The name of the Application or System that originally generated the Batch Interface Data File.

mdg:sourceSystemType

The technology category on which the Source System was hosted when the Batch Interface Data File was generated

mdg:sourceSystemOS

The Operating system on which the Source System was hosted when the Batch Interface Data File was generated

mdg:interfaceName

The name of the interface for which the Data File was generated. N.B. The InterfaceName and InterfaceVersion must uniquely identify the data file contents structure and its eventual destination(s).

mdg:interfaceVersion

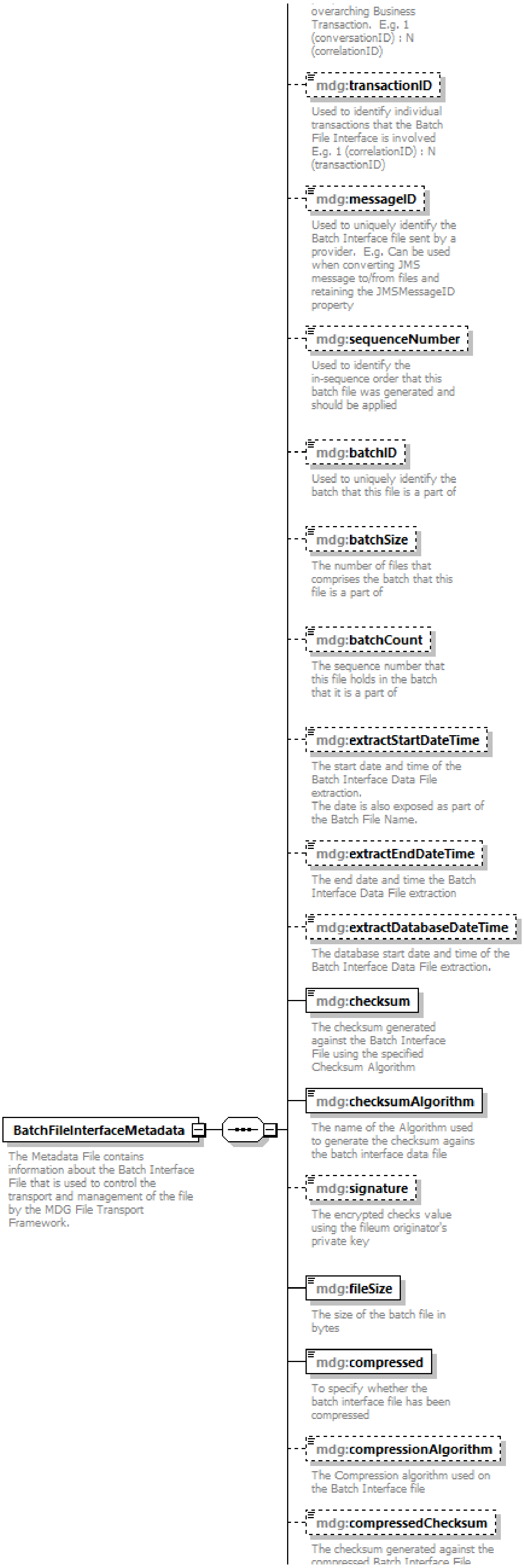
The version of the interface for which the Data File was generated. N.B. The InterfaceName and InterfaceVersion must uniquely identify the data file contents structure and its eventual destination(s).

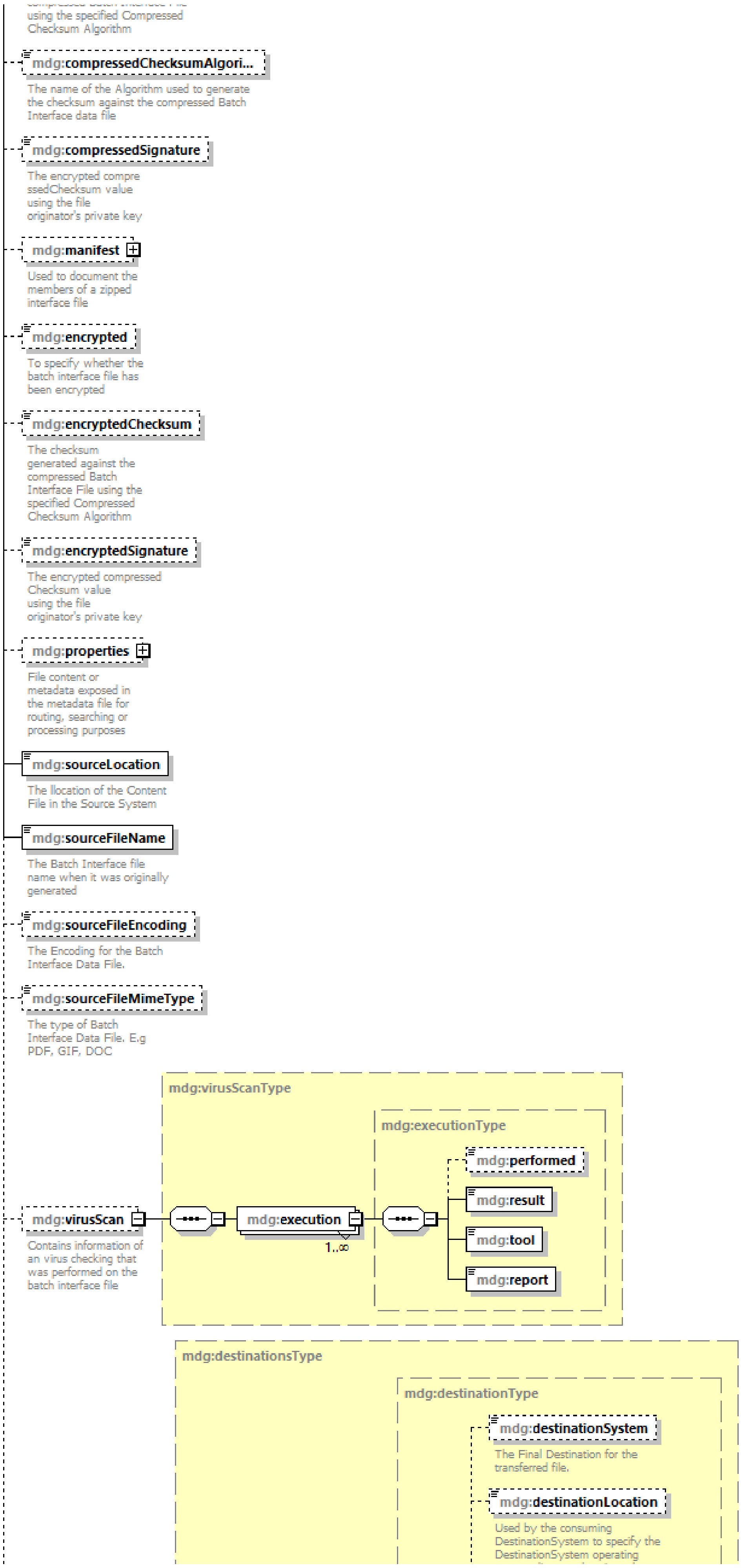
mdg:correlationID

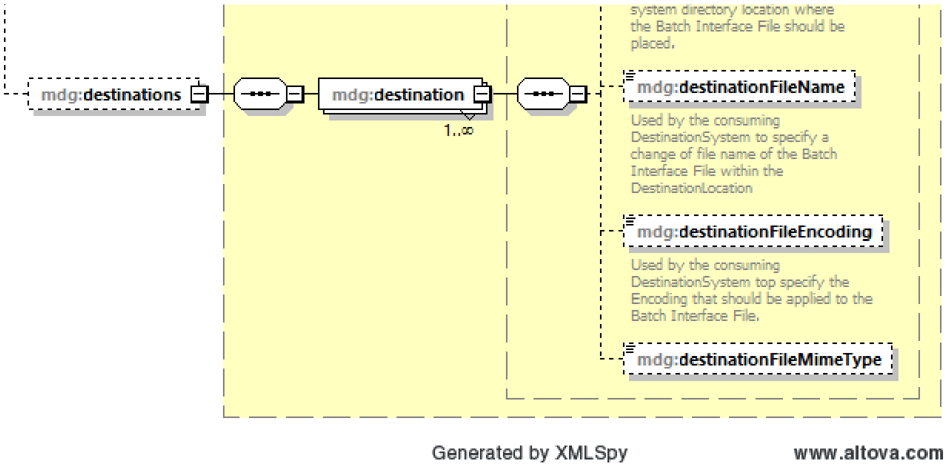
The unique identifier for each Batch Interface File journey through the Integration Layer so that a request can be matched to a response. N.B. If the correlationID is not supplied by the application that generated the Batch Interface File, it will be generated at the first opportunity by the Integration Layer as a UUID.

mdg:conversationID

Used to logically group a number of related Batch Interface Files from the perspective of an


overarching Business Transaction. E.g. 1 (conversationID) : N (correlationID)
mdg:transactionID
Used to identify individual transactions that the Batch File Interface is involved E.g. 1 (correlationID) : N (transactionID)
mdg:messageID
Used to uniquely identify the Batch Interface file sent by a provider. E.g. Can be used when converting JMS message to/from files and retaining the JMSMessageID property
mdg:sequenceNumber
Used to identify the in-sequence order that this batch file was generated and should be applied
mdg:batchID
Used to uniquely identify the batch that this file is a part of
mdg:batchSize
The number of files that comprises the batch that this file is a part of
mdg:batchCount
The sequence number that this file holds in the batch that it is a part of
mdg:extractStartDateTime
The start date and time of the Batch Interface Data File extraction.
The date is also exposed as part of the Batch File Name.
mdg:extractEndDateTime
The end date and time the Batch Interface Data File extraction
mdg:extractDatabaseDateTime
The database start date and time of the Batch Interface Data File extraction.
mdg:checksum
The checksum generated against the Batch Interface File using the specified Checksum Algorithm
mdg:checksumAlgorithm
The name of the Algorithm used to generate the checksum agains the batch interface data file
BatchFileInterfaceMetadata
The Metadata File contains information about the Batch Interface File that is used to control the transport and management of the file by the MDG File Transport Framework.
mdg:signature
The encrypted checks value using the fileum originator's private key
mdg:fileSize
The size of the batch file in bytes
mdg:compressed
To specify whether the batch interface file has been compressed
mdg:compressionAlgorithm
The Compression algorithm used on the Batch Interface file
mdg:compressedChecksum
The checksum generated against the compressed Batch Interface File


using the specified Compressed Checksum Algorithm
mdg:compressedChecksumAlgori...
The name of the Algorithm used to generate the checksum against the compressed Batch Interface data file
mdg:compressedSignature
The encrypted compre ssedChecksum value using the file originator's private key
mdg:manifest
Used to document the members of a zipped interface file
mdg:encrypted
To specify whether the batch interface file has been encrypted
mdg:encryptedChecksum
The checksum generated against the compressed Batch Interface File using the specified Compressed Checksum Algorithm
mdg:encryptedSignature
The encrypted compressed Checksum value using the file originator's private key
mdg:properties
File content or metadata exposed in the metadata file for routing, searching or processing purposes
mdg:sourceLocation
The llocation of the Content File in the Source System
mdg:sourceFileName
The Batch Interface file name when it was originally generated
mdg:sourceFileEncoding
The Encoding for the Batch Interface Data File.
mdg:sourceFileMimeType
The type of Batch Interface Data File. E.g PDF, GIF, DOC
mdg:virusScanType
mdg:executionType
mdg:performed
mdg:result
mdg:tool
mdg:report
mdg:virusScan
mdg:execution
1..∞
Contains information of an virus checking that was performed on the batch interface file
mdg:destinationsType
mdg:destinationType
mdg:destinationSystem
The Final Destination for the transferred file.
mdg:destinationLocation
Used by the consuming DestinationSystem to specify the DestinationSystem operating


system directory location where the Batch Interface File should be placed.
mdg:destinations
mdg:destination
1..∞
mdg:destinationFileName
Used by the consuming DestinationSystem to specify a change of file name of the Batch Interface File within the DestinationLocation
mdg:destinationFileEncoding
Used by the consuming DestinationSystem top specify the Encoding that should be applied to the Batch Interface File.
mdg:destinationFileMimeType
Generated by XMLSpy
www.altova.com

5. Naming Convention

Source File Name - The file from DIT must be named in one of the below format:
DIT_<Start Date - YYYYMMDD>-<End Date - YYYYMMDD>-<Timestamp - YYYYMMDDTHHMMSS>-EUFileSequence.XML
DIT_<Start Date - YYYYMMDD>-<End Date - YYYYMMDD>-<Timestamp - YYYYMMDDTHHMMSS>-EUMODFileSequence.XML
DIT_<Start Date - YYYYMMDD>-<End Date - YYYYMMDD>-<Timestamp - YYYYMMDDTHHMMSS>-NATIONALFileSequence.XML

Metadata File Name - DIT when creating the metadata file exactly as the source file with "_metadata" as a suffix.

DIT_<Start Date - YYYYMMDD>-<End Date - YYYYMMDD>-<Timestamp - YYYYMMDDTHHMMSS>-EUFileSequence_metadata.XML
DIT_<Start Date - YYYYMMDD>-<End Date - YYYYMMDD>-<Timestamp - YYYYMMDDTHHMMSS>-EUMODFileSequence_metadata.XML
DIT_<Start Date - YYYYMMDD>-<End Date - YYYYMMDD>-<Timestamp - YYYYMMDDTHHMMSS>-NATIONALFileSequence_metadata.XML

Destination File Name - MDG when pushing the file to QAM will retain the source file name. i.e. from the example above DIT_20181001-20180901212211-1234.XML. Since the file will be zipped and then base64 encoded the zip file name will be retained as the original file name i.e DIT_20181001-20180901212211-1234.encode.

6.Scope

1. Pull file from DIT FTP server
2. PUT file to CDS file store AWS S3 bucket
3. Send the TARIC3 file as a payload in a SOAP call to QAM
4. Only Taric XML file is in scope

7.Out Of Scope

1. Taric file schema validation
2. Taric EDIFACT file is out of scope
3. Email notification for error scenarios
4. MDG does not host CDS file store

8.Risks, Assumptions, Issues, Dependencies and Decisions

8.1 Risks

1. If the approved pattern from SDD V2.0 is not adopted by Tariff/bitzesty/DIT or if the pattern is modified in future then there is a risk of potential rework for MDG.

8.2 Assumptions

1. It is assumed that the source filename will be in one of the below formats:
   DIT_<Start Date - YYYYMMDD>-<End Date - YYYYMMDD>-<Timestamp -

YYYYMMDDTHHMMSS>-EUFileSequence_metadata.XML
DIT_<Start Date - YYYYMMDD>-<End Date - YYYYMMDD>-<Timestamp - YYYYMMDDTHHMMSS>-EUMODFileSequence_metadata.XML
DIT_<Start Date - YYYYMMDD>-<End Date - YYYYMMDD>-<Timestamp - YYYYMMDDTHHMMSS>-NATIONALFileSequence_metadata.XML

2. It is assumed that MDG will be required to PUT the taric file on to CDS File store S3 bucket location
3. It is assumed that the CDS file store folder structure will be as below:
   ToQAM
   Success
   Failure
4. **The SOAP call implementation conforms to QAM SDD CSD 128 Quota Allocation - V1.4 document**
5. The implementation conforms to the file transfer pattern defined in SDD V2.0
6. **Antivirus check:** Before placing the file on the DIT FTPS server a DIT micro service must scan the file and place the file only if it is virus free.
7. **Email notification and failure scenario management:** MDG are not required to send email notification on failure.

8.3 Issues

None Identified

8.4 Dependencies

1. MDG are dependent on DIT, QAM and project teams to review MDG API specification documents.

8.5 Decisions

1. CTO team has suggested that the email notification for error scnearios is not a recommended solution. Hence MDG would not be developing an email notification component until further confirmation/revised SDD from the project team.

9. TARIC3 file transfer scenarios and expected MDG behaviour

S.No	Scenario	Approach	Scenario Type (Success scenario, Error scenario)
1	TARIC3 File sent as normal	MDG interface (TAQ01) to move the file to CDS file store S3 bucket and will send the file as a payload in a SOAP to QAM.	Success scenario
2	TARIC3 File name skipped. i.e. Day1 – File sent Day2 – File Sent Day3 – Day4 file sent instead of Day 3	File names will be unique per day and there can be days with no data is sent. Action should be ignore and proceed as normal.	Exception Scenario
3	TARIC3 File not sent on a working day. i.e. Day1 – File sent Day2 – File Sent Day3 – File not sent Day 4 – File sent	No action required from MDG.	Exception Scenario

4	TARIC3 file checksum did not match to the checksum in the metadata file i.e. Day1 – File sent Day2 – File Sent Day3 – Checksum did not match	MDG will reject the TARIC3 file.	Error Scenario
5	TARIC3 File not sent on a National holiday or a bank holiday or a weekend. i.e. Day1 – File sent Day2 – File Sent Christmas – File not sent	Files are not generated on bank holidays/weekends. No action required from MDG.	Exception Scenario
6	Two files in one day.	There can be two or more files published in a single day, al files will have to be processed in sequence as normal.	Exception Scenario
7	Multiple files with same name	If there are multiple files with same name however it has corresponding metadata files then MDG will transfer the files to CDS file store S3 bucket and will send the file as a payload in a SOAP to QAM.	Exception Scenario
8	File transfer failure at DT/CT/AWS	MDG to move the file to the relevant failure folder	Error Scenario

Failure scenarios:

Following are the failure scenarios:

1. Job abruptly fails due to network glitch
2. Metadata file schema validation error
3. Checksum mismatch i.e. the checksum in the metadata file and the source file (taric or transaction) checksum are different
4. File name mismatch i.e. the source filename in the metadata XML and the actual filename are different
5. Source file type not XML
6. Failure Layer: scenario 1 and 3 at any stage of the file transfer (DT or CT)

Expected behaviour:
DIT FTPS Server: The source and the metadata files will be moved to failure folder on the DIT FTPS server
MDG DT and CT Servers: The source and the metadata files must be cleaned up as the files will have to be resubmitted (place the file on the FTPS server) via DIT
Notification: Not in scope need requirement from the programme. We should work with Vipul on failure logging using ELK
On success:
DIT FTPS Server: The source and the metadata files will be moved to success folder on the DIT FTPS server
MDG DT and CT Servers: The source and the metadata files must be cleaned up
Notification: Not in scope need requirement from the programme.

10. Validation Requirements

1. MDG will not validate the content of the TARIC file, i.e. taric file schema validation will not be performed by MDG
2. Checksum validation will be performed by MDG to ensure if the file is correctly transferred from the source location to the destination location
3. Metadata will be validated against the metadata schema

11. Security Requirements

1. MDG will perform a FTPS file transfer
2. SOAP call will be over HTTPS

12. Non-functional Requirements

1. Relevant firewalls requests must be raised to establish connectivity between DIT server and MDG environment.

13. Taric file example

DIT_20181001-20180901212211-1234.xml

14. Sample Metadata file

```
<?xml version="1.0" encoding="utf-8"?>
<BatchFileInterfaceMetadata
xmlns="http://www.hmrc.gsi.gov.uk/mdg/batchFileInterfaceMetadataSchema"
xmlns:vc="http://www.w3.org/2007/XMLSchema-versioning"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.hmrc.gsi.gov.uk/mdg/batchFileInterfaceMet
adataSchema
file:///C:/Users/7895902/Desktop/TAQ01/BatchFileInterfaceMetadata-1.0.6.
xsd">
 <sourceSystem>DIT</sourceSystem>
 <sourceSystemType>FTPS</sourceSystemType>
 <sourceSystemOS>RedHat_7.3</sourceSystemOS>
 <interfaceName>TAQ01</interfaceName>
 <interfaceVersion>V1.0</interfaceVersion>
 <correlationID>8452f702-a7e0-40aa-8d2f-052073d9ca88</correlationID>
 <conversationID>TODO</conversationID>
 <transactionID>TODO</transactionID>
 <messageID>0267c550-7c61-4dbe-812b-fe0f1bee90f7</messageID>
 <extractStartDateTime>2018-07-26T11:55:49Z</extractStartDateTime>
 <extractEndDateTime>2018-07-26T11:55:54Z</extractEndDateTime>
 <extractDatabaseDateTime>2018-07-26T11:55:54Z</extractDatabaseDateTime>
 <checksum>033839f93e730424800d924d4a9ca95e</checksum>
 <checksumAlgorithm>MD5</checksumAlgorithm>
 <fileSize>20992</fileSize>
 <compressed>false</compressed>
 <compressionAlgorithm>ZIP</compressionAlgorithm>
 <compressedChecksum>074dc93973877b3fdf43a7322e22a1ee</compressedChecksu
m>
 <compressedChecksumAlgorithm>MD5</compressedChecksumAlgorithm>
 <sourceLocation>DIT FTP Server</sourceLocation>
 <sourceFileName>DIT_20180726-20180726115549-TARICFileSequence.xml</sour
ceFileName>
 <sourceFileEncoding>UTF-8</sourceFileEncoding>
 <destinations>
  <destination>
   <destinationSystem>QAM</destinationSystem>
   <destinationLocation>QAM SOAP Call</destinationLocation>

<destinationFileName>DIT_20180726-20180726115549-TARICFileSequence.encod
e</destinationFileName>
   <destinationFileEncoding>BASE64</destinationFileEncoding>
  </destination>
 </destinations>
</BatchFileInterfaceMetadata>
```