

**Interface Control Document:**

**ICD DBS Managed File Transfer Service for eBulk**

**Product Reference: DBS\_EBU\_0310**

**Version : 2.0**

Date : 19/02/2024

Contents

1. Introduction 4

1.1 Document purpose 4

1.2 Scope 4

1.2.1 In scope 4

1.2.2 Out of scope 4

1.3 Glossary and abbreviations 5

1.3.1 Notational conventions 5

1.4 Reference Documents 6

2. Interface overview 7

2.1 Participating systems 7

2.1.1 DBS MFTS 7

2.1.2 Registered Bodies 7

2.2 Interface Purpose 7

3. Interface Details 8

3.1 Push method 8

3.2 Polling components 9

3.3 Pull method 9

3.4 File naming and directory structures 9

3.4.1 Inbound messages 10

3.4.2 Outbound messages 10

3.5 Handling errors 11

3.6 Files Transferred over the Internet or via the Public Services Network (PSN) 11

3.6.1 FTPS 11

3.6.2 FTPS Clients 11

3.6.3 Data types 12

3.6.4 User credentials 12

3.6.5 Maximum Message size 12

3.6.6 Connection details 12

3.6.7 Sending Files 12

3.6.8 Receiving Files 13

3.7 Additional Interface Details 13

3.7.1 Managed File Transfer 13

3.7.2 Case sensitivity 13

3.7.3 Restricted Information 14

3.7.4 Business Message Specification 14

3.7.5 Interchange Agreements 14

# Introduction

## Document purpose

This document describes the DBS eBulk interface between the DBS Managed File Transfer Service (MFTS) and Registered Bodies’ (RBs) in terms of the messages destined for the Disclosure & Barring Service (DBS). The eBulk interface described in this document provides for the exchange of files between the DBS MFTS and RBs systems using the File Transfer Protocol (FTP) over the internet and Public Services Network (PSN).

The document has a target audience of:

* DBS / Registered Body (RB) Project Teams
* implementers, developers and technical architects of the communicating endpoints
* testers of the interface
* readers who wish to gain a technical understanding of the interface

## Scope

### In scope

This document is concerned only with the interface between the DBS MFTS and Registered Bodies’ systems using the internet and Public Services Network. Only messages supported in the scope of the DBS e-Bulk project as defined in the DBS e-Bulk Interface *Business Process Document* [eBulk] are described by this document.

### Out of scope

This specification does not:

* cover “end-to-end” message exchanges, where communication between the DBS MFTS and RBs is part of a larger message exchange including other systems
* mandate the underlying technology platform of the endpoint implementations
* specify the manual administrative procedures that may need to be carried out when this interface is implemented as part of a specific system
* Specify the scalability, sizing, performance or reliability requirements of the DBS MFTS; these are documented within the delivery project documentation.

This document is organised in three sections, as follows:

| **Section** | **Contents** |
| --- | --- |
| 1. Introduction | Gives a brief introduction to the DBS MFTS interface |
| 2. Interface Overview | Covers background information about the interface endpoints and the requirements and constraints which drive the technical specification. |
| 3. Interface Details | Describes the FTP interface in terms of initiation of file transfer, directory structures and file naming conventions. |
| 4. Thru MFTS Configuration Details | Describes the internal configuration required by the DBS MFTS. This section does not need to be read by the DBS / RB audience. |

Table ‑: Document Organisation

## Glossary and abbreviations

### Notational conventions

The keywords "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC2119 [RFC2119].

| **Term** | **Meaning** |
| --- | --- |
| DBS | Disclosure & Barring Service. Refers to both the agency of the Home Office and to its IT programme, dependent on the context |
| FTP | File Transfer Protocol |
| FTPS | FTP secured with TLS |
| Glue Reply | Glue Reply are the global service integrator engaged by DBS to deploy and manage the Thru MFTS service on our behalf. |
| HTTP | Hypertext Transfer Protocol |
| ID | Identifier |
| IETF | Internet Engineering Task Force (http://www.ietf.org/) |
| IP | Internet Protocol |
| MFTS | Managed File Transfer Service |
| PSN | Public Services Network is the UK government's high-performance network, which helps public sector organisations work together, reduce duplication and share resources. |
| RB | Registered Body. A party authorised by the Disclosure and Barring Service to apply to the DBS for disclosures. |
| RFC | Request for comments maintained by the IETF Secretariat(http://www.ietf.org/rfc.html) |
| TCP | Transmission Control Protocol |
| Thru  | Thru is the service provider of our new Thru Managed File Transfer (MFT) service, which is a private cloud-based technology platform that allows DBS to reliably exchange electronic data between RB's and our processing systems in a secure way that meets compliance needs. |
| TLS | Transport Layer Security. A cryptographic protocol that provides secure communications over a computer network. |
| XML | Extensible Markup Language |

Table ‑: Glossary

## Reference Documents

| **Document ID** | **Document Title** | **Source** |
| --- | --- | --- |
| FTPMDI | DBS MFTS Message Delivery Interface Specification FTP | DBS |
| RFC2119 | Key words for use in RFCs to Indicate Requirement Levels. March 1997.(http://www.ietf.org/rfc/rfc2119.txt) | IETF |
| eBulk | e-Bulk Interface Business Process Document | DBS |
| BMS | e-Bulk Business Message Specification | DBS |

Table ‑: Reference Documents

# Interface overview

## Participating systems

### DBS MFTS

The DBS Managed File Transfer Service (MFTS) provides a common base for connectivity and information exchange between DBS and external organisations. The DBS MFTS allows systems based on different technology platforms to be integrated uniformly and offers a variety of messaging services such as transformation, reliability and routing so that this functionality is standardised across all external interfaces. The DBS MFTS is realised by the Thru service.

### Registered Bodies

Registered Bodies are those parties authorised by the Disclosure and Barring Service to submit applications for criminal records check and receive disclosure of relevant details.

The interface between Registered Bodies and the DBS MFTS forms part of a series of interactions known as the e-Bulk interface. More details may be found in the e-Bulk Interface *Business Process Document* [eBulk].

Each Registered Body has its own IT system and these IT systems will be different for each Registered Body. However, all RB systems connected to the DBS MFTS via the internet or Public Services Network will use the same interface, as described in this document, the associated *Business Process Document* [eBulk] and *Business Message Specification* [BMS].

## Interface Purpose

The messages in this interface are designed to support business processes as documented within the *e-Bulk Interface Business Process Document* [eBulk]. This does not preclude use of this same interface to support other processes.

The business level messages supported by the interface are defined in the eBulk Business Message Specification [BMS]. It is beyond the scope of this document to specify how the business messages should be created, populated with data and validated. However it is RECOMMENDED that message formatting is validated before leaving Registered Bodies’ system boundaries.

The table below summarises the messages sent between RB systems and the DBS MFTS.

| **Message type** | **Data Type** | **Message Name**  | **Description** | **Sender** | **Receiver** |
| --- | --- | --- | --- | --- | --- |
| CRB01 | XML/text | eBulkApplicationsBatch | Disclosure applications | RB | DBS MFTS |
| CRB02 | XML/text | eBulkApplicationsBatchRejection | Used to indicate an error condition (such as schema validation failure) encountered by DBS when processing the file. | DBS MFTS | RB |
| CRB03 | XML/text | eBulkApplicationReceiptsBatch | Application Acknowledgements  | DBS MFTS | RB |
| CRB04 | XML/text | eBulkResultsBatch | Results of Disclosure processing | DBS MFTS | RB |

Table ‑1 Business Messages

All business level data definitions are controlled and managed through the DBS’s *Business Message Specification* [BMS].

# Interface Details

The diagram below shows how the messages flow between the two systems.



Figure ‑1: Business Message Flow

All business messages transferred from RB systems to DBS MFTS and from DBS MFTS to RB systems will be transferred as files using the File Transfer Protocol (FTP) across the internet or Public Services Network (PSN).

Where the communication between the Managed File Transfer Service and external systems is over the internet, then FTPS will be used to secure the transfer/session. While the majority of information presented in this document is applicable to both the FTP and FTPS interfaces there are technological and behavioural differences between the two interfaces.

In this specification, File Transfer is used as an asynchronous delivery mechanism. Each message is actually a file transferred from a storage device on one system to a storage device on another system. Unlike a request-response protocol such as HTTP, there is no concept of a response to an FTP file transfer.

There are a number of methods for moving messages from one system to another using FTP, however for external systems using the eBulk solution CRB01 messages will be pushed to the Managed File Transfer Service. The CRB02, CRB03 and CRB04 response messages will be pushed to the Managed File Transfer Service by the DBS system.

## Push method

For all business messages originating from RB systems, the Push method SHALL be used (i.e. the RB system initiates the transfer with the FTP put command).

In this method, the sending system proactively sends a file to the receiving system using the FTP STOR command and renames it using a convention (described below) to indicate that transfer is complete. From the sender’s point of view, the file has been submitted for processing as soon as the file has finished transfer and has been renamed. However, the receiver has to inspect the relevant directory to know that the file has arrived. There may therefore be a significant time gap between the sender’s submission end time and the receiver’s processing start time.

Once the file’s arrival has been noticed by the receiver, it is usually copied to a subsystem via some mechanism for further processing, which is out of scope for this document. However, the receiver MUST ensure that the file is not processed again. This specification does not define how the receiver is to achieve this, as local housekeeping procedures may be in place. However, it is RECOMMENDED that as soon as the file has been sent for processing, the receiver renames the file with a different extension such as “.proc”, and optionally moves it to an archive directory or allows a housekeeping process to remove and archive it at a later date.

Note that the local FTP client implementation may wrap the STOR command and associated FTP PORT commands into a single call, often called “put”.

There is no specific schedule for the transfer of files between RBs and the DBS MFTS and therefore the DBS MFTS should expect RBs to connect to the DBS MFTS at any time. Note that there may be more than one file per day.

## Polling components

Because the sender is using the Push method, the receiver needs a mechanism for becoming aware that a message has arrived, for example a component which polls a directory at configured intervals to see if there are any new files. If the receiver is using the Pull method, it needs a component which can poll the sender’s directory at configured intervals; however, the polling component is able to send incoming messages for processing as soon as they are transferred.

## Pull method

For all business messages being sent to RB systems by the DBS MFTS, the RB system SHALL use the Pull method (i.e. Registered Body initiates the transfer with the FTP get command).

## File naming and directory structures

This section specifies the FTP directory and file names on the DBS MFTS FTP server for each message type. The sender of the business message MUST conform to the following rules for file naming. Note that there are no blank spaces in the filenames below and underscore characters are used to separate parts of the filename. Files originating from RBs have the prefix “RB\_” while those originating from DBS have the prefix “CRB\_”. The DBS MFTS will retain the original file name for the files that it delivers to the DBS. For files that are delivered to the DBS MFTS by the RB the DBS MFTS will make the files available to DBS with the original RB file names. Note that duplicate files will be rejected by the DBS eBulks systems. Files that have the same name are considered to be duplicates.



Figure ‑2: RB Directory Structure

### Inbound messages

The MFTS shall implement the directory structure shown in Figure 3-2 for incoming messages delivered using the Push method. The MFTS shall have a file storage directory (the FTP root directory) whose sub-directories can be accessed using FTP.

For each sending system, there shall be a file storage directory located directly under the FTP root directory, which is named after the sending system.

The sending system directory SHALL contain a sub-directory named “in”.

For each business message type that the sending system sends, there shall be a sub-directory of “in” which is named after the business message type. The sender must transfer messages to the appropriate directory based on the business message type of the file.

Sending systems shall only be given write access to the incoming sending system directory and its subdirectories.

### Outbound messages

Sending systems SHALL implement the following directory structures for outgoing messages delivered using the Pull method (where the receiving system uses FTP GET on the sending system):

The sending system SHALL have a file storage directory (the FTP root directory) whose sub-directories can be accessed using FTP.

For each receiving system, there SHALL be a directory located directly under the FTP root directory, which is named after the receiving system, in this case it is the <RB System ID>.

The sending system directory SHALL contain a sub-directory named “out”.

For each business message type that the receiving system expects to receive, there SHALL be a sub-directory “out” which is named after the business message type. The sending system SHALL place files in the appropriate directory based on the business message type of the file.

Receiving systems using the Pull method SHALL be given permissions sufficient to allow them to retrieve files and create “.flg” files used to indicate that the file was successfully processed.

## Handling errors

There are several types of error which may occur when using this delivery mechanism including the following:

* Network transmission errors where data is corrupted or lost in transit
* FTP protocol errors
* Operational request errors which occur when the MFTS attempts to import a file sent to it by a external organisation due to errors in the file name or similar problem
* Errors which occur on the FTP outbound interface when the MFTS attempts to construct the filename for an outbound file
* Business level errors which occur when the ultimate destination system fails to correctly process a business message

All systems MUST monitor their FTP processing and log any errors encountered for manual resolution by the system administrator. Details of the manual resolution process MUST be defined in the Interchange Agreement (IA).

## Files Transferred over the Internet or via the Public Services Network (PSN)

All business messages transferred over the internet or via PSN from RB systems to the MFTS and from the MFTS to RB systems will be transferred as files using FTPS. RB systems will be responsible for initiating all file transfer requests.

### FTPS

FTP with SSL/TLS (FTPS) is an enhancement of the industry standard FTP. It uses the same FTP commands and protocol but over secure sockets, i.e. SSL/TLS in order to provide secure file transfers.

### FTPS Clients

To connect to the DBS Managed File Transfer Service an FTP client is needed.

The FTP client must be able to support:

* explicit FTPS file transfers [AUTH TLS mode (Explicit FTPS)];
* the use of Client Authentication Certificates;
* Passive FTP connections with TLS 1.2;
* Binary Mode

The RB systems in order to protect data in transit to and Thru must have all connections encrypted using TLS 1.2, using one of the supported and appropriate Ciphers.

### Data types

If the business message file is an XML document, the sender MUST set the FTP representational type to be IMAGE using the TYPE command with the “I” (IMAGE) parameter value to avoid conversion of the document to an incorrect codepage. N.B.: FTB clients often use “binary” as the type name rather than IMAGE.

If the business message file is an ASCII document, the sender MAY set the FTP type to ASCII using the TYPE command with the A (ASCII) parameter value, or MAY use IMAGE as above. ASCII is the default type, but if documents of various types are being transferred by the same FTP client in sequence, care should be taken.

If the business message file is an EBCDIC document, the sender MUST set the FTP type to EBCDIC using the TYPE command with the E (EBCDIC) parameter value.

### User credentials

Receiving systems must provision a user for the use of the sending system and inform the appropriate operations team of the user credentials. This user will be used by the sending system for logging on to the FTP serving on the receiving system. Details will be stored in the ICD for the specific interface.

FTP user names MUST contain only upper- and lower-case alphabet characters (A-Z and a-z) and numeric characters 0-9, as some operating systems do not allow punctuation characters such as underscores (\_) because of security concerns.

It is RECOMMENDED that user names are between 6 and 8 characters long, to allow for consistency between operating systems. Some older UNIX systems, for example, do not allow user names to be longer than 8 characters, whereas other operating systems allow 20 or more.

### Maximum Message size

The MFTS has a restriction on the size of messages it can process which is dependent on the interface used and the content of the message. The maximum message size is currently 4 MB.

### Connection details

The details of the environmental URLs and public certificates for the internet and PSN for the production, pre-production, and development and test environments are provided.

### Sending Files

The directory structure on the DBS MFTS FTP server for files sent from RBs to the DBS MFTS will have the structure:

***/DBSMFT/<RB System ID>/in/<Message Type>***

An example of this input directory is */XXX /in/CRB01* where XXX is an RB System ID.

Files placed into the input directory by RBs will be named as follows:

***RB\_< DBS RB Reference Number>\_<Message Type>\_<Message ID >\_<DateTime>.xml***

An example of a file placed in this directory is *RB\_01234567890\_CRB01\_91012345\_20070102030505.xml*

### Receiving Files

The directory structure on the DBS MFTS FTP server for files destined for the RBs will have the format:

***/DBSMFT/<RB System ID>/out/<Message Type>***

An example of this output directory is */XXX/RB\_01234567890/out/CRB02* where XXX is an RB System ID.

Files placed into the output directory by the DBS MFTS will be named as follows:

***CRB\_ <DBS RB Reference Number>\_<message type>\_<Message ID>\_<DateTime>.xml***

An example of a file placed in this directory is *CRB\_01234567890\_CRB02\_78012345\_20070102030505.xml*

The <> components in the above structures have the meanings defined in the table below.

| **Component** | **Meaning** |
| --- | --- |
| <RB System ID> | A System ID allocated by DBS. Each RB will be allocated a unique value. The list of RB System ID values currently allocated can be found in the DBS CRM ICD. Each RB will be notified of their System ID and FTP account details separately. |
| <Message Type> | A 5-digit code identifying the type of message being sent, as defined in Table 2-1 on page 7 above. i.e. CRB01, CRB02, CRB03 and CRB04 |
| <DBS RB Reference Number> | DBS provided identifier (11 digits) that uniquely identifies an RB system. The DBS System ID and DBS RB Reference Number will be the same. |
| <Message ID> | An 8 digit identifier (such as an incrementing number) provided by the originator of the message for the logical grouping of business elements in the batch (e.g. a batch of eBulkApplications). See the DBS Business Message Specification for further details on Message ID. |
| <DateTime> | Date/time at which the file was written in the format YYYYMMDDHHMMSS. |

Table ‑2: Key to directory and file name components

## Additional Interface Details

### Managed File Transfer

This section applies to the Push method only.

A summary of the key points applicable to RB interface follows:

* Message files being delivered to the DBS MFTS (FTP put) should initially be created with a “-“ suffix. Once they have been successfully delivered they can be renamed removing the “-“ suffix.
* Internal processes and external systems that poll for files will ignore files with a “-“ suffix.
* Once a message file has been successfully downloaded from the out directory, it can either be DELETED or a flag file can be created in the out directory for each retrieved file (FTP get). On a periodic basis the flag file and associated message file will deleted from the directory by housekeeping processes within the DBS MFTS.
* As an example, a file named  “*RB\_01234567890\_CRB01\_91012345\_20070102030505.xml*”
MUST be transferred using the name:
  “*RB\_01234567890\_CRB01\_91012345\_20070102030505.xml-*”. When the transfer is complete, the file is renamed by the sender to
 “*RB\_01234567890\_CRB01\_91012345\_20070102030505.xml*”.

### Case sensitivity

Directory and file names MUST be treated as case-sensitive by both sender and receiver, as this specification does not mandate the operating system or storage type of either. Specific interfaces MAY further require that all lower- or all upper-case is used.

### Restricted Information

The FTP User Name, RB System ID, Password, DBS RB Reference Number, IP address and other details related to the security of the connection between the DBS MFTS and RBs are not defined in this document. These details will be documented in separate restricted communications that are specific to each RB.

### Business Message Specification

The business level messages supported by the interface are defined in the eBulk Business Message Specification [BMS]. It is RECOMMENDED that message formatting is validated before leaving Registered Bodies’ system boundaries as this will not be performed by the DBS MFTS. However, invalid messages will be rejected by the DBS eBulk system.

### Interchange Agreements

An Interchange Agreement (IA) will need to be created in order to record respective intentions and expectations of each party involved in the interface. Interchange Agreements are the responsibility of the data owner (DBS) and define the high level information management requirements of connection between two parties in terms of:

• The purpose of data sharing

• The parties to the data sharing

• The business processes necessary to support data sharing

• The volumetrics / Frequency of data sharing

• The delivery mechanism

• The service management

• Any Standards / Policies / Legislation (as applicable) that apply

• Contact Details of key stakeholders

END OF DOCUMENT