DLP MESH Setup User Guide

How to send DLP data using MESH

Last updated October 2021

Information and technology
for better health and care

Contents

Glossary of Terms

Abbreviation	Meaning
API	Application Programming Interface
CSV	Comma Separated Values
DLP	Data Landing Portal
DTS	Data Transfer Service
GP	General Practice
ID	Identification
MESH	Messaging Exchange for Social Care and Health
NDSD	National Data Services Development
NHS	National Health Service
NSD	National Service Desk
ODS	Organisation Data Services

Data Landing Portal

The Data Landing Portal (DLP) seeks to replace current local email data transfers between providers (Data Senders) and commissioners (Data Recipients) and support standardisation by using a secure, centrally managed system. DLP enables Data Recipients to set up Data Specifications, against which incoming data from providers is validated.

MESH

MESH (Messaging Exchange for Social Care and Health) is the main messaging service used across health and social care. It is the replacement for the preceding Data Transfer Service (DTS) and works on the Spine infrastructure. MESH is used to transfer electronic messages directly from one application to another, enabling different organisations to communicate securely.

MESH can be accessed using the MESH client, or through systems that have been integrated using the Application Programming Interface (API). Further information regarding implementing MESH using an API is available on the MESH API webpages. In most cases systems suppliers or IT administrators will handle the set-up of MESH for each organisation, and users should contact their IT department with any further questions.

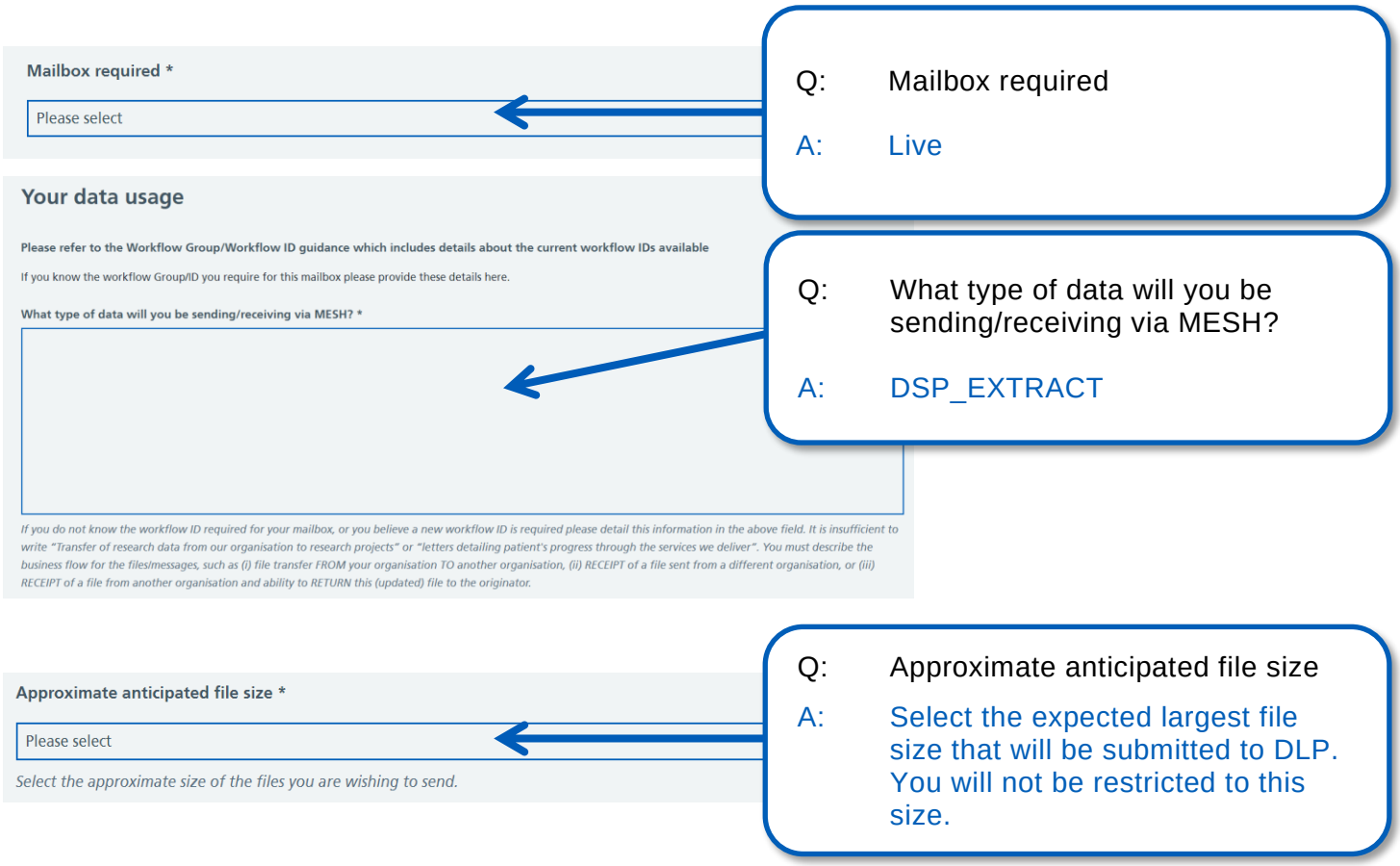
N.B. while this guide is about how to send data to DLP using MESH, MESH is a separate service supported by a different team within NHS Digital.

How to set up the MESH client for data transfers

Step 1 – Request a MESH Mailbox

The first step in setting up the MESH client is to request a MESH mailbox. This is done by completing the MESH application form. If your organisation already has a MESH mailbox set-up, please proceed to Step 2.

To ensure smooth completion of the application, the required responses to certain questions are provided below. The following screenshots identify these questions and the required NHS Digital response or advice is highlighted in blue.



Once you have submitted the MESH application form please proceed to step 2 if you are implementing the MESH client, if you are using an API please proceed to Step 3.

Step 2 – Install MESH client

To install the MESH client, download the MESH client installation pack and the MESH client installation guidance from the MESH and MOLES technical documentation and file downloads webpage. The MESH client installation guidance outlines the process for installation, including pre-requisite checks, and guidance for different operating systems (Windows and Linux). Please refer to the MESH webpage for more information or contact the National Service Desk if you experience any issues with MESH installation.

Step 3 – Client Certificate

Importantly, within the MESH client installation guide you will find information on applying for the Client (Keystore) Certificate. The certificate is needed for authentication purposes and is essential for you to successfully complete your installation. Due to security reasons, we cannot apply for one on your behalf.

The link below will further guide you through the process to be followed:

How to use the MESH certificate enrolment tool

Step 4 – Send Confirmation

Having successfully set up a MESH mailbox with MESH Client or set up an API, please send confirmation, including the Organisation Data Service (ODS) code, and MESH mailbox ID for your organisation to the Service Management team via the National Service Desk e-mail address, specifying that you want to set up MESH to make submissions to DLP. The DLP team will then complete the configuration changes to enable you to send automated DLP data transfers via MESH.

Setting Up Control Files

Each submission to DLP via MESH requires a file pairing, comprising of:

- A **.dat** file - The data/ payload to be transferred, which in the case of DLP is the **ANSI** (**Windows-1252**) encoded CSV submission file. It has to be renamed with the **.dat** extension.
- A .**ctl** file - This is the control file. It contains the information required to identify, transmit and audit the data file. The control file will need to be updated with your specification ID and filename. This is explained in more detail below.

Each file pairing will need to share the same exact name, with the **.ctl** and **.dat** extension added respectively.

The control file **MUST** match the format of the following example, with the items highlighted in **bold-red** replaced with your own organisation's details, as per the **accompanying note** (found to the side of the field in question). **Without these changes your file will not be successfully delivered and processed.**

Example OUT Ctl File

```
<DTSControl>
  <Version>1.0</Version>
  <AddressType>ALL</AddressType>
  <MessageType>Data</MessageType>
  <WorkflowId>DSP_EXTRACT</WorkflowId>
  <From_ESMTP>XXXX@dts.nhs.uk</From_ESMTP> *This needs to be your mailbox ID
  <To_ESMTP>X26HC010@dts.nhs.uk</To_ESMTP>
  <From_DTS>XXXX</From_DTS> *This needs to be your mailbox ID
  <To_DTS>X26HC010</To_DTS>
  <Subject>b2ac587139c34646bc5ca6cbfb600cf5</Subject> *The Specification ID to be used for the
submission – you can get this from your data recipient if you are unsure what it is
  <LocalId>my_submission.csv</LocalId> *The filename for the submission (this MUST be a .csv or .csv.gz file)
  <Compress>Y</Compress>
  <Encrypted>N</Encrypted>
  <AllowChunking>Y</AllowChunking>
</DTSControl>
```

Sending Files to DLP via MESH

Once the two files are appropriately named and configured, they will need to be placed together in your MESH **OUT**-folder (.dat first if moved individually).

After placing in the **OUT**-folder the user will need to run the MESH Client. This will then send the files to the MESH Server. If the MESH Client is already running the files will be sent when the next MESH Client poll happens (this is configurable within each MESH Client).

Once transferred by MESH, the outbound control file is amended with the transaction details (as per the example below) and placed within the client **SENT**-folder, along with the paired outbound submission .dat file.

Example SENT Ctl File

```
<DTSControl>
  <Version>1.0</Version>
  <AddressType>ALL</AddressType>
  <MessageType>Data</MessageType>
  <From_DTS> XXXX </From_DTS>
  <To_DTS>X26HC010</To_DTS>
  <From_ESMTP>XXXX@dts.nhs.uk</From_ESMTP>
  <To_ESMTP>X26HC010@dts.nhs.uk</To_ESMTP>
  <Subject>b2ac587139c34646bc5ca6cbfb600cf5</Subject>
  <LocalId>my_submission.csv</LocalId>
  <DTSId>20180919102252052139_A39867</DTSId>
  <Compress>Y</Compress>
  <Encrypted>N</Encrypted>
  <WorkflowId>DSP_EXTRACT</WorkflowId>
  <IsCompressed>Y</IsCompressed>
  <AllowChunking>Y</AllowChunking>
  <StatusRecord>
    <DateTime>20180919112341</DateTime>
    <Event>TRANSFER</Event>
    <Status>SUCCESS</Status>
    <StatusCode>00</StatusCode>
    <Description>Data Transfer success confirmation</Description>
  </StatusRecord>
</DTSControl>
```

This indicates the file has been successfully sent to MESH. Once the outbound message has been received by DLP's MESH mailbox, DLP will send a response back via MESH, to

the sender's **IN**-folder, as confirmation of delivery and whether the submission was successful.

DLP File Validation Returns

On receipt of the sender's submission by DLP, the sender's control file details will be checked to ensure that:

- Workflow ID is correct
- Specification ID is correct
- Filename is a csv/ csv.gz
- Sender's MESH mailbox is in our ref data

The submission file will be checked to ensure:

- File size is within limits – Currently 4GB for csv and 800MB for csv.gz
- The file structure is correct, as per the Specification
- Data fields conform to any validation rules set in the Specification

The outcome of these checks will be sent back to the sender's **IN**-folder, again as a **.ctl** and **.dat** file pair. The .dat file is the **'submission_result.json'**, which will state whether the submission was successful or not (with the reason given for failure).

Example IN Ctl File for Submission Response

```
<DTSControl>
  <Version>1.0</Version>
  <AddressType>ALL</AddressType>
  <MessageType>Data</MessageType>
  <From_DTS>X26HC010</From_DTS>
  <To_DTS>XXXX</To_DTS>
  <From_ESMTP>X26HC010@dts.nhs.uk</From_ESMTP>
  <To_ESMTP>XXXX@dts.nhs.uk</To_ESMTP>
  <Subject>SubmissionResultDLPv1</Subject>
 <LocalId>submission_result.json</LocalId>
  <DTSId>20180919102312390939_0B1CFF</DTSId>
  <PartnerId></PartnerId>
  <Compress>N</Compress>
  <Encrypted>N</Encrypted>
  <WorkflowId>DSP_EXTRACT</WorkflowId>
  <ProcessId></ProcessId>
  <DataChecksum></DataChecksum>
  <IsCompressed>N</IsCompressed>
  <StatusRecord>
    <DateTime>20180919112312</DateTime>
    <Event>TRANSFER</Event>
    <Status>SUCCESS</Status>
    <StatusCode>00</StatusCode>
    <Description>Transferred to recipient mailbox</Description>
  </StatusRecord>
</DTSControl>
```

Submissions that have successfully passed the first 5 bulleted checks, as listed above, will appear in the DLP User Interface (UI), in the same manner they would had the files been manually uploaded to the system.

Should any validation rules failures be found they will be confirmed in the returned **submission_result.json**, indicated by a **validation_counts** item showing as 1 or more:

Example

"**validation_counts**": {

"errors": 0, **Validation failures which will result in file rejection – submission failure*

"no_action": 5, **Validation failures with no resulting action*

"warnings": 1 **Validation failures meriting warning, but not causing submission failure*

},

In this case, you will need to login into the DLP UI to download the validation report for your submission, which will fully detail the cause for any failures.

Submission result JSON return samples – All cases

The following case samples display the full range of return results you can expect to receive back, depending upon the success or failure of the submission and the type of errors (if any) found with the file.

1. Successful Upload with no errors


```
{
        "submission_date": "2018-10-09T08:45:53.901710+00:00",
        "submission_id": "57e16e7e41194505b94f1d18cea416fa",
        "submission_mesh_id": "20180919102252052139_A39867”,
        "file_name": "my_submission.csv",
        "spec_id": "b2ac587139c34646bc5ca6cbfb600cf5",
        "status": "success",
        "validation_errors": [],
        "workflow_id": "DSP_EXTRACT"
}
```

2. Upload failed because of CSV errors


```
{
        "submission_date": "2018-10-09T08:45:53.901710+00:00",
               "submission_id": "57e16e7e41194505b94f1d18cea416fa",
        "submission_mesh_id": "20180919102252052139_A39867",
        "file_name": "my_submission.csv",
        "spec_id": "b2ac587139c34646bc5ca6cbfb600cf5",
        "status": "failed",
        "validation_errors": [
               "Column A was 'Alpha' (expected 'Name')",
               "Column D is not within agreed specification",
               "Row 2 has 7 columns (expected 6)"
        ],
        "workflow_id": "DSP_EXTRACT"
}
```


3. Successful upload against a specification with Data Field Checking (DFC) rules


```
{
        "submission_date": "2018-10-09T08:45:53.901710+00:00",
        "submission_id": "57e16e7e41194505b94f1d18cea416fa",
        "submission_mesh_id": "20180919102252052139_A39867",
        "file_name": "my_submission.csv",
        "spec_id": "b2ac587139c34646bc5ca6cbfb600cf5",
        "status": "success",
        "validation_errors": [],
        "workflow_id": "DSP_EXTRACT"
}
```

4. Upload failed against specification with DFC rules (all failures are because of 'File rejection' errors)


```
{
        "submission_date": "2018-10-09T08:45:53.901710+00:00",
               "submission_id": "57e16e7e41194505b94f1d18cea416fa",
        "submission_mesh_id": "20180919102252052139_A39867",
        "file_name": "my_submission.csv",
        "spec_id": "b2ac587139c34646bc5ca6cbfb600cf5",
        "status": "failed",
        "validation_counts": {
               "errors": 7,
               "no_action": 0,
               "warnings": 0
        },
        "validation_errors": [],
        "workflow_id": "DSP_EXTRACT"
}
```


5. Upload Successful against specification with DFC rules (all failures are because of 'Warnings')


```
{
        "submission_date": "2018-10-09T08:45:53.901710+00:00",
               "submission_id": "57e16e7e41194505b94f1d18cea416fa",
        "submission_mesh_id": "20180919102252052139_A39867",
        "file_name": "my_submission.csv",
        "spec_id": "b2ac587139c34646bc5ca6cbfb600cf5",
        "status": "success",
        "validation_counts": {
               "errors": 0,
               "no_action": 0,
               "warnings": 7
        },
        "validation_errors": [],
        "workflow_id": "DSP_EXTRACT"
}
```

6. Upload Successful against specification with DFC rules even if the file has DFC errors (because failure action is 'No action')


```
{
        "submission_date": "2018-10-09T08:45:53.901710+00:00",
                "submission_id": "57e16e7e41194505b94f1d18cea416fa",
                "submission_mesh_id": "20180919102252052139_A39867",
                "file_name": "my_submission.csv",
        "spec_id": "b2ac587139c34646bc5ca6cbfb600cf5",
        "status": "success",
        "validation_counts": {
                "errors": 0,
                "no_action": 7,
                "warnings": 0
        },
        "validation_errors": [],
        "workflow_id": "DSP_EXTRACT"
}
```


7. Upload failed against a specification with DFC rules because of a combination of errors, warnings, no action.


```
{
        "submission_date": "2018-10-09T08:45:53.901710+00:00",
                "submission_id": "57e16e7e41194505b94f1d18cea416fa",
                "submission_mesh_id": "20180919102252052139_A39867",
                "file_name": "my_submission.csv",
        "spec_id": "b2ac587139c34646bc5ca6cbfb600cf5",
        "status": "failed",
        "validation_counts": {
                "errors": 3,
                "no_action": 2,
                "warnings": 2
        },
        "validation_errors": [],
        "workflow_id": "DSP_EXTRACT"
}
```

8. Upload failed because of Invalid specification id


```
{
        "submission_date": "2018-10-09T08:45:53.901710+00:00",
                "submission_id": "57e16e7e41194505b94f1d18cea416fa",
         "submission_mesh_id": "20180919102252052139_A39867",
        "file_name": "my_submission.csv",
        "reason": "Specification ID invalid",
        "spec_id": "b2ac587139c34646bc5ca6cbfb600cf5",
        "status": "failed",
        "workflow_id": "DSP_EXTRACT"
}
```


9. Upload failed because of Inactive specification


```
{
        "submission_date": "2018-10-09T08:45:53.901710+00:00",
                "submission_id": "57e16e7e41194505b94f1d18cea416fa",
                "submission_mesh_id": "20180919102252052139_A39867",
                "file_name": "my_submission.csv",
        "reason": "Specification is not active",
        "spec_id": "b2ac587139c34646bc5ca6cbfb600cf5",
        "status": "failed",
        "workflow_id": "DSP_EXTRACT"
}
```


10. Upload failed because Specification ID is not associated with the Sender organization


```
{
        "submission_date": "2018-10-09T08:45:53.901710+00:00",
                "submission_id": "57e16e7e41194505b94f1d18cea416fa",
                "submission_mesh_id": "20180919102252052139_A39867",
                "file_name": "my_submission.csv",
        "reason": "Specification ID not associated with sender organisation",
        "spec_id": "b2ac587139c34646bc5ca6cbfb600cf5",
        "status": "failed",
        "workflow_id": "DSP_EXTRACT"
}
```

11. Upload failed because file size exceeded the maximum allowed size


```
{
        "submission_date": "2018-10-09T08:45:53.901710+00:00",
                "submission_id": "57e16e7e41194505b94f1d18cea416fa",
                "submission_mesh_id": "20180919102252052139_A39867",
                "file_name": "my_submission.csv",
        "reason": "Upload exceeded maximum allowed size",
        "spec_id": "b2ac587139c34646bc5ca6cbfb600cf5",
        "status": "failed",
        "workflow_id": "DSP_EXTRACT"
}
```


Contacts and Useful Links

National Service Desk – Telephone 0300 303 5035 or ssd.nationalservicedesk@nhs.net

NHS Digital DLP web pages for further information and supporting materials

MESH and MOLES technical documentation and file downloads - Library of downloadable resources

To be added to the distribution list for regular updates to service suppliers and users, email mesh@nhs.net