

GV File

FIMS API Guide



Version History

Date	Version	Release by	Reason for Changes
08/12/2012	0.01	Steve Dabner	Document originated (1 st proposal)
18/02/2013	0.02	Steve Dabner	Added new commands
12/07/2013	0.03	Steve Dabner	Additional elements and notes added
08/11/2013	0.04	Steve Dabner	Modified output file name elements
13/11/2013	0.05	Steve Dabner	Added estimated Completion Duration
05/09/2014	1.0	Andy Gingell	Migrated into to new document template
09/09/2015	1.1	Andy Gingell	Rebranded to SAM GV File
20/01/2016	1.2	Greg Emerson	Completed rebrand, new user extensions see 3.1.5 note 4
01/07/2016	1.3	Simon Collins	Added profile version wildcard information
21/12/2016	1.4	Simon Collins	Added target duration and marker information for Kronos jobs
09/05/2018	1.5	Simon Collins	Updated profile version wildcard information.
10/10/2018	1.6	Jon Metcalf	GV rebrand

Table of Contents

1. Definitions, Acronyms and Abbreviations	4
2. Introduction	5
2.1 Purpose	5
2.2 References	5
2.3 Server URLs	5
3. Supported Messages	6
3.1 Client Sends these Messages	6
3.1.1 queryJob (by ID).....	6
3.1.2 queryJob (by filter)	6
3.1.3 manageJob	7
3.1.4 manageQueue	8
3.1.5 transformRequest.....	9
3.2 Server Responds with these Messages	18
3.2.1 queryJobResponse	18
3.2.2 manageJobResponse	23
3.2.3 manageQueueResponse	25
3.2.4 transformAck	26

List of Tables

Table 1: Table of Terminology	4
Table 2: Manage Job Current Supported Commands	7
Table 3: Manage Queue Current Supported Commands	8
Table 4: TransformRequest Output File Patterns	17

1. Definitions, Acronyms and Abbreviations

Table 1: Table of Terminology

Term	Definition
FIMS	Framework for Interoperable Media Services
AMWA	Advanced Media Workflow Association
EBU	European Broadcasting Union
SOA	Service Oriented Architecture
MPP	Media Processing Platform
SOAP	Simple Object Access Protocol
WSDL	Web services description language
XSD	XML Schema Document

2. Introduction

2.1 Purpose

This document describes the FIMS SOAP messages that are used to interface with the GV File server application. These messages are used by clients to add jobs to the job queue, and check the status of jobs and manage jobs and queues.

In the following sections the SOAP XML messages for each operation and response are shown. Within the messages the fields that are used by the GV File server are highlighted in yellow. Note that many of the fields within the messages are optional. The optional elements of the messages may be omitted by clients even if they are shown below to clarify the placement of the used message fields.

2.2 References

[FIMS Documentation](#)

[FIMS Implementation](#)

2.3 Server URLs

The GV File server listens for messages on a single network port. The path that the SOAP messages are sent to determines which of the FIMS services receives incoming messages.

For example, if the GV File server is listening on address example.com port 35061 then the SOAP messages should be directed as follows:-

- Transform media service messages go to: <http://example.com:35061/transform/job>
- Transform media status service messages go to: <http://example.com:35061/transform/status>
- Transform media notification service messages go to: <http://example.com:35061/transform/notify>

3. Supported Messages

3.1 Client Sends these Messages

3.1.1 queryJob (by ID)

Use this to enquire after the status of a particular job. Send this via the transform media status service.

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  soap:encodingStyle="http://schemas.xmlsoap.org/soap/soap-encoding"
  xmlns:base="http://base.fims.tv" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:soap="soap">
  <soapenv:Header/>
  <soapenv:Body>
    <base:queryJobRequest xsi:type="base:QueryJobRequestByIDType">
      <base:jobInfoSelection>all</base:jobInfoSelection>
      <!--Optional:-->
      <base:ExtensionGroup>
      </base:ExtensionGroup>
      <base:ExtensionAttributes/>
      <base:jobID>urn:smp:ul:12345678.12345678.12345678.12345678</base:jobID>
    </base:queryJobRequest>
  </soapenv:Body>
</soapenv:Envelope>
```

The resource ID of the job being queried

3.1.2 queryJob (by filter)

Use this to ask for the status of a set of jobs as specified by the filter criteria. Send this via the transform media status service.

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:base="http://base.fims.tv" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soapenv:Header/>
  <soapenv:Body>
    <base:queryJobRequest xsi:type="base:QueryJobRequestByFilterType">
      <base:jobInfoSelection>all</base:jobInfoSelection>
      <base:ExtensionGroup>
      </base:ExtensionGroup>
      <base:ExtensionAttributes/>
      <base:listFilter>
        <base:fromDate>2013-02-14T16:28:00.115</base:fromDate>
        <base:toDate>2013-02-15T16:28:00.115</base:toDate>
        <base:includeQueued>true</base:includeQueued>
        <base:includeActive>true</base:includeActive>
        <base:includeFinished>false</base:includeFinished>
      </base:listFilter>
    </base:queryJobRequest>
  </soapenv:Body>
</soapenv:Envelope>
```

List Filter = filter criteria for this status query. All criteria are optional

```

    <base:includeFailed>false</base:includeFailed>
    <base:maximumResults>64</base:maximumResults>
  </base:listFilter>
</base:queryJobRequest>
</soapenv:Body>
</soapenv:Envelope>

```

3.1.3 manageJob

Use this to manage an individual job. Send this via the transform media status service.

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:base="http://base.fims.tv">
  <soapenv:Header/>
  <soapenv:Body>
    <base:manageJobRequest>
      <base:jobID>urn:smp:ul:060E2B34.04010105.050E0A01.01004F4B</base:jobID>
      <base:jobCommand>modifyPriority</base:jobCommand>
      <base:priority>high</base:priority>
      <base:ExtensionGroup>
      </base:ExtensionGroup>
      <base:ExtensionAttributes/>
    </base:manageJobRequest>
  </soapenv:Body>
</soapenv:Envelope>

```

Identify job to be managed

Command to be undertaken

New priority (if required)

Table 2: Manage Job Current Supported Commands

Name	Supported / Not Supported
Modify Priority	supported
Cancel Job	supported
Pause Job	not supported
Resume Job	not supported
Restart Job	not supported
Stop Job	not supported
Cleanup Job	not supported

3.1.4 manageQueue

Use this to manage a job queue. Send this via the transform media status service.

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:base="http://base.fims.tv">
  <soapenv:Header/>
  <soapenv:Body>
    <base:manageQueueRequest>
      <base:queueID>0</base:queueID>
      <base:queueCommand>lock</base:queueCommand>
      <base:ExtensionGroup>
      </base:ExtensionGroup>
      <base:ExtensionAttributes/>
    </base:manageQueueRequest>
  </soapenv:Body>
</soapenv:Envelope>

```

Table 3: Manage Queue Current Supported Commands

Name	Supported / Not Supported
Status	supported
Clear	supported
Lock	supported
UnLock	supported
Start	supported
Stop	supported

3.1.5 transformRequest

Use this to request a new job. Normally the job would be added to the job queue by the receiver. Send this via the transform media service.

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:tran="http://transformmedia.fims.tv" xmlns:base="http://base.fims.tv"
xmlns:des="http://description.fims.tv">
  <soapenv:Header/>
  <soapenv:Body>
    <tran:transformRequest>
      <transformJob>
        <base:resourceID?></base:resourceID>
        <base:revisionID?></base:revisionID>
        <base:location?></base:location>
        <base:resourceCreationDate?></base:resourceCreationDate>
        <base:resourceModifiedDate?></base:resourceModifiedDate>
        <base:notifyAt>
          <base:replyTo?></base:replyTo>
          <base:faultTo?></base:faultTo>
        </base:notifyAt>
        <base:ExtensionGroup>
        </base:ExtensionGroup>
        <base:ExtensionAttributes/>
        <base:status?></base:status>
        <base:statusDescription?></base:statusDescription>
        <base:serviceProviderJobID?></base:serviceProviderJobID>
        <base:queueReference><!--Additional elements here--></base:queueReference>
        <base:tasks><!--Additional elements here--></base:tasks>
        <base:operationName?></base:operationName>
        <base:bmObjects>
          <base:bmObject>
            <base:resourceID?></base:resourceID>
            <base:revisionID?></base:revisionID>
            <base:location?></base:location>
            <base:resourceCreationDate?></base:resourceCreationDate>
            <base:resourceModifiedDate?></base:resourceModifiedDate>
            <base:notifyAt>
              <base:replyTo?></base:replyTo>
              <base:faultTo?></base:faultTo>
            </base:notifyAt>
            <base:ExtensionGroup>
              <userExtension>

```

The resource ID of the job being requested. Optional, if blank Viticulture will allocate a resource ID.

If supplied, it must be unique to this job.

Clients requiring 'fault' and 'result' notifications should put their endpoints for notifications

```

<input><n9:timecode>00:10:00:00</n9:timecode></input>
<output><n9:timecode>00:20:00:00</n9:timecode></output>
<inputEnabled>true</inputEnabled>
<outputEnabled>true</outputEnabled>
<multiplier>2</multiplier>
<targetDuration>00:00:30:00</targetDuration>
<markerFile></markerFile>
<markers>
  <marker>
    <name>advert 1 </name>
    <inTc>01:03:10;00</inTc>
    <outTc>01:03:12;00</outTc>
    <durationTc>00:00:02;00</durationTc>
  </marker>
  <marker>
    <name>Advert 2</name>
    <inTc>01:03:15;00</inTc>
    <outTc>01:03:18;10</outTc>
    <durationTc>00:00:03;10</durationTc>
  </marker>
</markers>
</userExtension>
</base:ExtensionGroup>
<base:ExtensionAttributes/>
<base:bmContents>
  <base:bmContent>
    <base:resourceID?</base:resourceID>
    <base:revisionID?</base:revisionID>
    <base:location?</base:location>
    <base:resourceCreationDate?</base:resourceCreationDate>
    <base:resourceModifiedDate?</base:resourceModifiedDate>
    <base:notifyAt>
      <base:replyTo?</base:replyTo>
      <base:faultTo?</base:faultTo>
    </base:notifyAt>
    <base:ExtensionGroup>
  </base:ExtensionGroup>
<base:ExtensionAttributes/>
<base:bmContentFormats>
  <base:bmContentFormat>
    <base:resourceID?</base:resourceID>
    <base:revisionID?</base:revisionID>
    <base:location?</base:location>

```

In/Out point parameters

Processing Speed multiplier

Target Duration

Marker File Location

Marker Parameters

```

<base:resourceCreationDate>?</base:resourceCreationDate>
<base:resourceModifiedDate>?</base:resourceModifiedDate>
<base:notifyAt>
  <base:replyTo>?</base:replyTo>
  <base:faultTo>?</base:faultTo>
</base:notifyAt>
<base:ExtensionGroup>
</base:ExtensionGroup>
<base:ExtensionAttributes/>
<base:bmEssenceLocators>
  <base:bmEssenceLocator>
    <base:resourceID>?</base:resourceID>
    <base:revisionID>?</base:revisionID>
    <base:location>?</base:location>
    <base:resourceCreationDate>?</base:resourceCreationDate>
    <base:resourceModifiedDate>?</base:resourceModifiedDate>
    <base:notifyAt>
      <base:replyTo>?</base:replyTo>
      <base:faultTo>?</base:faultTo>
    </base:notifyAt>
    <base:ExtensionGroup>
    </base:ExtensionGroup>
    <base:ExtensionAttributes/>
    <base:storageType typeLabel="" typeDefinition="" typeLink="">
      ?</base:storageType>
    <base:locatorInfo>?</base:locatorInfo>
    <base:containerMimeType typeLabel="" typeDefinition="" typeLink="">
      ?</base:containerMimeType>
  </base:bmEssenceLocator>
</base:bmEssenceLocators>
<base:formatCollection>
  <base:videoFormat><!--Additional elements here--></base:videoFormat>
  <base:audioFormat><!--Additional elements here--> </base:audioFormat>
  <base:dataFormat><!--Additional elements here--></base:dataFormat>
  <base:containerFormat><!--Additional elements here--></base:containerFormat>
</base:formatCollection>
<base:duration>
  <base:timecode>?</base:timecode>
  <base:normalPlayTime>?</base:normalPlayTime>
  <base:editUnitNumber editRate="" factorNumerator="" factorDenominator="">
    ?</base:editUnitNumber>
</base:duration>
<base:hash>

```

Source file name

```

    <base:hashFunction typeLabel="" typeDefinition="" typeLink="">
      ?</base:hashFunction>
    <base:value>cid:1252272354731</base:value>
  </base:hash>
  <base:packageSize>?</base:packageSize>
  <base:mimeType typeLabel="" typeDefinition="" typeLink="">?</base:mimeType>
</base:bmContentFormat>
</base:bmContentFormats>
  <base:descriptions><!--Additional elements here--></base:descriptions>
</base:bmContent>
</base:bmContents>
</base:bmObject>
</base:bmObjects>
<base:priority>?</base:priority>
<base:startJob/>
<base:finishBefore>?</base:finishBefore>
<base:estimatedCompletionDuration>?</base:estimatedCompletionDuration>
<base:currentQueuePosition>?</base:currentQueuePosition>
<base:jobStartedTime>?</base:jobStartedTime>
<base:jobElapsedTime>?</base:jobElapsedTime>
<base:jobCompletedTime>?</base:jobCompletedTime>
<base:processed>
  <base:percentageProcessedCompleted>?</base:percentageProcessedCompleted>
</base:processed>
<profiles>
  <transformProfile name="" description="">
    <base:resourceID>?</base:resourceID>
    <base:revisionID>?</base:revisionID>
    <base:location>?</base:location>
    <base:resourceCreationDate>?</base:resourceCreationDate>
    <base:resourceModifiedDate>?</base:resourceModifiedDate>
    <base:notifyAt>
      <base:replyTo>?</base:replyTo>
      <base:faultTo>?</base:faultTo>
    </base:notifyAt>
    <base:ExtensionGroup>
    </base:ExtensionGroup>
    <base:ExtensionAttributes/>
    <base:service>
      <base:resourceID>?</base:resourceID>
      <base:revisionID>?</base:revisionID>
      <base:location>?</base:location>
      <base:resourceCreationDate>?</base:resourceCreationDate>

```

Job priority

Name and type of profile

See Note 3 below for more details

Profile file name

See Notes 3 & 3.1 below for more details

Reserved for internal use

```

    <base:resourceModifiedDate>?</base:resourceModifiedDate>
    <base:notifyAt>
      <base:replyTo>?</base:replyTo>
      <base:faultTo>?</base:faultTo>
    </base:notifyAt>
    <base:ExtensionGroup>
  </base:ExtensionGroup>
  <base:ExtensionAttributes/>
  <base:providerName>?</base:providerName>
  <base:providerEndPoint>?</base:providerEndPoint>
  <base:serviceDescription><!--Additional elements here--></base:serviceDescription>
</base:service>
<transformAtom>
  <base:videoFormat><!--Additional elements here--></base:videoFormat>
  <base:audioFormat><!--Additional elements here--></base:audioFormat>
  <base:containerFormat><!--Additional elements here--></base:containerFormat>
  <base:ExtensionGroup>
</base:ExtensionGroup>
  <base:ExtensionAttributes/>
</transformAtom>
  <transferAtom>
    <base:destination>?</base:destination>
    <base:ExtensionGroup>
  </base:ExtensionGroup>
  <base:ExtensionAttributes/>
  </transferAtom>
  <outputFileNamePattern>?</outputFileNamePattern>
</transformProfile>
</profiles>
</transformJob>
</tran:transformRequest>
</soapenv:Body>
</soapenv:Envelope>

```

See Note 3.2 below for more details

Destination folder name

Destination output filename pattern
See Table 4.

Note 1. – there could be a list of BMOjects (with essence locations) if there are multiple input files

Note 2. – there will usually be a list of profiles with different descriptions (e.g solution, source etc). At the moment a configuration profile and a solution profile are expected.

Note 3. – Three parameters must be supplied as part of the transformProfile element.

These are:-

name – this is the descriptive name of the profile (e.g “Alchemist File”)

description – this says what type of profile file is being listed here. Currently this should be the (exact) string “solutionProfile” or “configurationProfile” as appropriate.

location - this specifies the location of profile file. This location is relative to the server directory that can be accessed by clients (/downloads). This location can be either a full URL or it can be a relative path.

For example if the server is listening on address *example.com* port *35061* then the solution profile location may be:

http://example.com:35061/downloads/products/Alchemist_XF/1.0.0.0/alchemyXFSolution.xml

Similarly the configuration profile location may be:

http://example.com:35061/downloads/products/Alchemist_XF/1.0.0.0/user_profiles/config_profile.1.xml

Note 3.1 – In the example above, the solution version is 1.0.0.0 (highlighted yellow). In order to prevent this version having to be updated each time a system is upgraded it is possible to replace the version number with the version wildcard text ‘current’. This allows the downstream services to substitute the version wildcard text (‘current’) with the current solution version number and will allow continued operation after upgrades.

Also it can be seen that the configuration profile is at version 1 (highlighted green). This version can also be replaced by the ‘current’ wildcard, which will substitute the wildcard text with the most current configuration profile version.

Modified solution profile location using wildcard:

http://example.com:35061/downloads/products/Alchemist_XF/current/alchemyXFSolution.xml

Modified configuration profile location using wildcards:

http://example.com:35061/downloads/products/Alchemist_XF/current/user_profiles/config_profile.current.xml

Note 3.2 – If the transform profile’s description is ‘solutionProfile’ then the ‘transferAtom’ section should be removed in its entirety, to prevent duplicate ‘destination’ fields being submitted.

Alternatively,

Change directory to:

C:\Program Files\SAM\clientAccess\products\Alchemist_XF\downloads\1.0.x.x

The file is: alchemistXFSolution.xml

Note 4 – There is a new user extension group called ‘userExtension’. This extension carries information for the in/out point parameters, processing speed multiplier, target duration, marker file location and marker parameter information. The following examples show the entries for the differing fields.

If the ‘userExtension’ is omitted, jobs will continue to be processed at up to 1 x real-time, and from start of material to end.

Example 1: Shows both in and out points enabled. An input point of 10 (frames) and an output point of 20 (frames). Also the processing speed (multiplier) is set to 2 (up to 2 x real-time processing)

```
<userExtension>
  <input><n9:editUnitNumber editRate="0" factorNumerator="0"
  factorDenominator="0">10</n9:editUnitNumber></input>
  <output><n9:editUnitNumber editRate="0" factorNumerator="0"
  factorDenominator="0">20</n9:editUnitNumber></output>
  <inputEnabled>true</inputEnabled>
  <outputEnabled>true</outputEnabled>
  <multiplier>2</multiplier>
</userExtension>
```

Example 2: Shows both in and out points enabled. An input point of 00:10:00:00 (timecode) and an output point of 00:20:00:00 (timecode). Also the processing speed (multiplier) is set to 2 (up to 2 x real-time processing)

```
<userExtension>
  <input><n9:timecode>00:10:00:00</n9:timecode></input>
  <output><n9:timecode>00:20:00:00</n9:timecode></output>
  <inputEnabled>true</inputEnabled>
  <outputEnabled>true</outputEnabled>
  <multiplier>2</multiplier>
</userExtension>
```

Example 3: Shows both in and out points enabled. An input point of 00:11:00 (11 minutes) and an output point of 00:22:00 (22 minutes). Also the processing speed (multiplier) is set to 2 (up to 2 x real-time processing)

```
<userExtension>
  <input><n9:normalPlayTime>00:11:00</n9:normalPlayTime></input>
  <output><n9:normalPlayTime>00:22:00</n9:normalPlayTime></output>
  <inputEnabled>true</inputEnabled>
  <outputEnabled>true</outputEnabled>
  <multiplier>2</multiplier>
</userExtension>
```

Example 4: Neither the in point or out point have been enabled so there are no in/out point parameters in the userExtension. The processing speed (multiplier) is set to 2 (up to 2 x real-time processing)

```
<userExtension>
  <multiplier>2</multiplier>
</userExtension>
```

Example 5: Shows a target duration of 30 seconds (for Kronos jobs).

```
<userExtension>
  <targetDuration>00:00:30:00</targetDuration>
</userExtension>
```

Example 6: Shows a (Kronos) target duration of 1 hour and a marker file found at location: `//Share/source/markerFiles/example.csv` (for Kronos jobs).

```
<userExtension>
  <targetDuration>01:00:00:00</targetDuration>
  <markerFile>//Share/source/markerFiles/example.csv</markerFile>
</userExtension>
```

Example 7: Shows a target duration of 1 hour, no specified marker file location, but explicit marker parameters (for Kronos jobs).

```
<userExtension>
  <targetDuration>01:00:00:00</targetDuration>
  <markerFile></markerFile>
  <markers>
    <marker>
      <name>advert 1 </name>
      <inTc>01:03:10;00</inTc>
      <outTc>01:03:12;00</outTc>
      <durationTc>00:00:02;00</durationTc>
    </marker>
    <marker>
      <name>Advert 2</name>
      <inTc>01:03:15;00</inTc>
      <outTc>01:03:18;10</outTc>
      <durationTc>00:00:03;10</durationTc>
    </marker>
  </markers>
</userExtension>
```

Further notes on file locations and profile files will be provided in a separate document.

When making a job request the output filename is automatically constructed from the destination folder and the output filename pattern elements of the transform profile SOAP element. The output filename pattern can contain plain text such as "myfile.mxf" and or some special strings that are substituted at the time of starting the job. The special strings are shown in **Error! Reference source not found.**

For example if the destination folder is set `c:\outputDir` and the output pattern is set to `Converted_%Source%` then if the source is `myfile.mxf` then the output file will be `c:\outputDir\Converted_myfile.mxf`. If no filename pattern is specified a default constructed file name is built.

Table 4: TransformRequest Output File Patterns

Pattern Content	Meaning
%Date%	The current date in UTC format
%Format%	A Descriptive name of video/audio formats
%Source%	Source file name
%Profile%	Profile name
%Increment%	An auto incrementing identifier (a number)

3.2 Server Responds with these Messages

3.2.1 queryJobResponse

This is the answer to a queryJob message. Received via the transform media status service.

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:base="http://base.fims.tv" xmlns:des="http://description.fims.tv">
  <soapenv:Header/>
  <soapenv:Body>
    <base:queryJobResponse>
      <base:jobs>
        <base:job>
          <base:resourceID?</base:resourceID>
          <base:revisionID?</base:revisionID>
          <base:location?</base:location>
          <base:resourceCreationDate?</base:resourceCreationDate>
          <base:resourceModifiedDate?</base:resourceModifiedDate>
          <base:notifyAt>
            <base:replyTo?</base:replyTo>
            <base:faultTo?</base:faultTo>
          </base:notifyAt>
          <base:ExtensionGroup>
          </base:ExtensionGroup>
          <base:ExtensionAttributes/>
          <base:status?</base:status>
          <base:statusDescription?</base:statusDescription>
          <base:serviceProviderJobID?</base:serviceProviderJobID>
        </base:job>
        <base:queueReference>
          <base:resourceID>0</base:resourceID>
          <base:revisionID xsi:nil="true"/>
          <base:location xsi:nil="true"/>
          <base:resourceCreationDate xsi:nil="true"/>
          <base:resourceModifiedDate xsi:nil="true"/>
          <base:notifyAt>
            <base:replyTo xsi:nil="true"/>
            <base:faultTo xsi:nil="true"/>
          </base:notifyAt>
          < base:ExtensionGroup xsi:nil="true"/>
          <base:ExtensionAttributes xsi:nil="true"/>
          <base:status>started</base:status>
          <base:statusDescription xsi:nil="true"/>
        </base:queueReference>
      </base:jobs>
    </base:queryJobResponse>
  </soapenv:Body>
</soapenv:Envelope>

```

The resource ID of the job whose status is being reported

Notification endpoint

The status of the job

Job failure descriptive messages

The identifier used by the server to identify this job in its database

The identifier used by the server to identify the queue the job is in.

Queue status information

```
<base:length>4</base:length>
<base:availability>true</base:availability>
<base:estimatedTotalCompletionDuration xsi:nil="true"/>
<base:jobs xsi:nil="true"/>
</base:queueReference>
<base:tasks/>
<base:operationName?></base:operationName>
<base:bmObjects>
  <base:bmObject>
    <base:resourceID?></base:resourceID>
    <base:revisionID?></base:revisionID>
    <base:location?></base:location>
    <base:resourceCreationDate?></base:resourceCreationDate>
    <base:resourceModifiedDate?></base:resourceModifiedDate>
    <base:notifyAt>
      <base:replyTo?></base:replyTo>
      <base:faultTo?></base:faultTo>
    </base:notifyAt>
    <base:ExtensionGroup>
  </base:ExtensionGroup>
  <base:ExtensionAttributes/>
  <base:bmContents>
    <base:bmContent>
      <base:resourceID?></base:resourceID>
      <base:revisionID?></base:revisionID>
      <base:location?></base:location>
      <base:resourceCreationDate?></base:resourceCreationDate>
      <base:resourceModifiedDate?></base:resourceModifiedDate>
      <base:notifyAt>
        <base:replyTo?></base:replyTo>
        <base:faultTo?></base:faultTo>
      </base:notifyAt>
      <base:ExtensionGroup>
    </base:ExtensionGroup>
    <base:ExtensionAttributes/>
    <base:bmContentFormats>
      <base:bmContentFormat>
        <base:resourceID?></base:resourceID>
        <base:revisionID?></base:revisionID>
        <base:location?></base:location>
        <base:resourceCreationDate?></base:resourceCreationDate>
        <base:resourceModifiedDate?></base:resourceModifiedDate>
        <base:notifyAt>
```

```

    <base:replyTo?></base:replyTo>
    <base:faultTo?></base:faultTo>
</base:notifyAt>
<base:ExtensionGroup>
  <Profile>
    <location?></location>
    <name?></name>
    <description?></description>
    <destination?></destination>
    <outputPattern?>< outputPattern >
    <outputFilePath?>< outputFilePath >
  </Profile>
</base:ExtensionGroup>
<base:ExtensionAttributes/>
<base:bmEssenceLocators>
  <base:bmEssenceLocator>
    <base:resourceID?></base:resourceID>
    <base:revisionID?></base:revisionID>
    <base:location?></base:location>
    <base:resourceCreationDate?></base:resourceCreationDate>
    <base:resourceModifiedDate?></base:resourceModifiedDate>
    <base:notifyAt>
      <base:replyTo?></base:replyTo>
      <base:faultTo?></base:faultTo>
    </base:notifyAt>
    <base:ExtensionGroup>
    </base:ExtensionGroup>
    <base:ExtensionAttributes/>
    <base:storageType typeLabel="" typeDefinition="" typeLink="">
      ?</base:storageType>
    <base:locatorInfo?></base:locatorInfo>
    <base:containerMimeType typeLabel="" typeDefinition="" typeLink="">
      ?</base:containerMimeType>
  </base:bmEssenceLocator>
</base:bmEssenceLocators>
<base:formatCollection> <!--Additional elements here-->
</base:formatCollection>
<base:duration>
  <base:timecode?></base:timecode>
  <base:normalPlayTime?></base:normalPlayTime>
  <base:editUnitNumber
    editRate=""
    factorNumerator=""

```

GV extension

Location = profile location

Name = profile name

Description = profile description

Destination = output folder

outputPattern – output file pattern

outputFilePath = actual output name

See Note 4 below for more details

The source file name?

```

        factorDenominator="?">
        ?</base:editUnitNumber>
    </base:duration>
    <base:hash>
        <base:hashFunction typeLabel="?" typeDefinition="?" typeLink="?">
            ?</base:hashFunction>
        <base:value>cid:723637806768</base:value>
    </base:hash>
    <base:packageSize>?</base:packageSize>
    <base:mimeType typeLabel="?" typeDefinition="?" typeLink="?">
        ?</base:mimeType>
    </base:bmContentFormat>
</base:bmContentFormats>
<base:descriptions>
    <base:description>
        <base:resourceID>?</base:resourceID>
        <base:revisionID>?</base:revisionID>
        <base:location>?</base:location>
        <base:resourceCreationDate>?</base:resourceCreationDate>
        <base:resourceModifiedDate>?</base:resourceModifiedDate>
        <base:notifyAt>
            <base:replyTo>?</base:replyTo>
            <base:faultTo>?</base:faultTo>
        </base:notifyAt>
        <base:ExtensionGroup>
        </base:ExtensionGroup>
        <base:ExtensionAttributes/>
        <des:bmContentDescription xml:lang="?">
            <!--Additional elements here-->
        </des:bmContentDescription>
    </base:description>
</base:descriptions>
</base:bmContent>
</base:bmContents>
</base:bmObject>
</base:bmObjects>
<base:priority>?</base:priority>
<base:startJob/>
<base:finishBefore>?</base:finishBefore>
<base:estimatedCompletionDuration>?</base:estimatedCompletionDuration>
<base:currentQueuePosition>?</base:currentQueuePosition>
<base:jobStartedTime>?</base:jobStartedTime>
    <base:jobElapsedTime>?</base:jobElapsedTime>

```

The job priority

Estimated time until running job completes

The jobs current queue position

The jobs start/end time

```

    <base:jobCompletedTime?></base:jobCompletedTime>
  <base:processed>
    <base:percentageProcessedCompleted?></base:percentageProcessedCompleted>
  </base:processed>
</base:job>
</base:jobs>
<base:notReportedResultsNumber?></base:notReportedResultsNumber>
<base:ExtensionGroup>
</base:ExtensionGroup>
<base:ExtensionAttributes/>
</base:queryJobResponse>
</soapenv:Body>
</soapenv:Envelope>

```

The percentage of the job completed

If a max number of results were set, how many were not reported

Note 4: There could be a list of jobs reported, as above, within a queryJobResponse

Note 5: The GV extension group adds some additional status reporting to the default FIMS query response. This extension group contains the following elements

Location = profile location as per Note 3 above

Name = profile name as per Note 3 above

Description = profile description as per Note 3 above

Destination = output folder this is the same as the destination field supplied in the job request transform profile and it contains the name of the folder in which the output file will be placed.

outputPattern – output file pattern this is the same as the output file name pattern as supplied in the job request transform profile see Table 4 for details

outputFilePath = this is the final calculated complete output filename. This field will be empty until the job has started as it is calculated by the system at that point.

3.2.2 manageJobResponse

This is the answer to a manageJob message. Received via the transform media status service.

```
<soap:Envelope soap:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/" xmlns:soap-
enc="http://schemas.xmlsoap.org/soap/encoding/" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
```

```
<soap:Body>
```

```
<base:manageJobResponse xmlns:n1="http://base.fims.tv">
```

```
<base:job>
```

```
<base:resourceID>urn:smp:ul:060E2B34.04010105.050E0A01.0100377D</base:resourceID>
```

```
<base:revisionID xsi:nil="true"/>
```

```
<base:location xsi:nil="true"/>
```

```
<base:resourceCreationDate xsi:nil="true"/>
```

```
<base:resourceModifiedDate xsi:nil="true"/>
```

```
<base:notifyAt>
```

```
<base:replyTo xsi:nil="true"/>
```

```
<base:faultTo xsi:nil="true"/>
```

```
</base:notifyAt>
```

```
<base:ExtensionGroup xsi:nil="true"/>
```

```
<base:ExtensionAttributes xsi:nil="true"/>
```

```
<base:status>queued</base:status>
```

```
<base:statusDescription></base:statusDescription>
```

```
<base:serviceProviderJobID>80</base:serviceProviderJobID>
```

```
<base:queueReference>
```

```
<base:resourceID>0</base:resourceID>
```

```
<base:revisionID xsi:nil="true"/>
```

```
<base:location xsi:nil="true"/>
```

```
<base:resourceCreationDate xsi:nil="true"/>
```

```
<base:resourceModifiedDate xsi:nil="true"/>
```

```
<base:notifyAt>
```

```
<base:replyTo xsi:nil="true"/>
```

```
<base:faultTo xsi:nil="true"/>
```

```
</base:notifyAt>
```

```
<base:ExtensionGroup xsi:nil="true"/>
```

```
<base:ExtensionAttributes xsi:nil="true"/>
```

```
<base:status>started</base:status>
```

```
<base:statusDescription xsi:nil="true"/>
```

```
<base:length>4</base:length>
```

```
<base:availability>true</base:availability>
```

```
<base:estimatedTotalCompletionDuration xsi:nil="true"/>
```

```
<base:jobs xsi:nil="true"/>
```

The resource ID of the job being managed

Notification endpoint

The status of the job

Job failure description messages

The identifier used by the server to identify this job in its database

The identifier used by the server to identify the queue the job is in

Queue status information

```
</base:queueReference>
<base:tasks xsi:nil="true"/>
<base:operationName xsi:nil="true"/>
<base:bmObjects xsi:nil="true"/>
<base:priority>high</base:priority>
<base:startJob xsi:nil="true"/>
<base:finishBefore xsi:nil="true"/>
<base:estimatedCompletionDuration xsi:nil="true"/>
<base:currentQueuePosition>0</base:currentQueuePosition>
<base:jobStartedTime xsi:nil="true"/>
<base:jobElapsedTime xsi:nil="true"/>
<base:jobCompletedTime xsi:nil="true"/>
<base:processed>
  <base:percentageProcessedCompleted>0</base:percentageProcessedCompleted>
</base:processed>
</base:job>
<base:ExtensionGroup xsi:nil="true"/>
<base:ExtensionAttributes xsi:nil="true"/>
</base:manageJobResponse>
</soap:Body>
</soap:Envelope>
```

The job priority

Estimated time until running job completes

The jobs current queue position

The jobs start/end time

The percentage of the job completed

3.2.3 manageQueueResponse

This is the answer to a manageQueue message. Received via the transform media status service.

```
<soap:Envelope soap:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/" xmlns:soap-
enc="http://schemas.xmlsoap.org/soap/encoding/" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soap:Body>
    <base:manageQueueResponse xmlns:n1="http://base.fims.tv">
      <base:queue>
        <base:resourceID>0</base:resourceID>
        <base:revisionID xsi:nil="true"/>
        <base:location xsi:nil="true"/>
        <base:resourceCreationDate xsi:nil="true"/>
        <base:resourceModifiedDate xsi:nil="true"/>
        <base:notifyAt>
          <base:replyTo xsi:nil="true"/>
          <base:faultTo xsi:nil="true"/>
        </base:notifyAt>
        <base:ExtensionGroup xsi:nil="true"/>
        <base:ExtensionAttributes xsi:nil="true"/>
        <base:status>locked</base:status>
        <base:statusDescription xsi:nil="true"/>
        <base:length>2</base:length>
        <base:availability>true</base:availability>
        <base:estimatedTotalCompletionDuration xsi:nil="true"/>
        <base:jobs xsi:nil="true"/>
      </base:queue>
      <base:ExtensionGroup xsi:nil="true"/>
      <base:ExtensionAttributes xsi:nil="true"/>
    </base:manageQueueResponse>
  </soap:Body>
</soap:Envelope>
```

The identifier used by the server to identify the queue that the job is in.

Queue status information

3.2.4 transformAck

This is the answer to a transformJob message. Received via the transform media service.

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:tran="http://transformmedia.fims.tv" xmlns:base="http://base.fims.tv" xmlns:des="http://description.fims.tv">
```

```
<soapenv:Header/>
```

```
<soapenv:Body>
```

```
<tran:transformAck>
```

```
<transformJob>
```

```
<base:resourceID?</base:resourceID>
```

The resource ID of the job whose job request is being acknowledged

```
<base:revisionID?</base:revisionID>
```

```
<base:location?</base:location>
```

```
<base:resourceCreationDate?</base:resourceCreationDate>
```

```
<base:resourceModifiedDate?</base:resourceModifiedDate>
```

```
<base:notifyAt>
```

```
<base:replyTo?</base:replyTo>
```

Notification endpoint

```
<base:faultTo?</base:faultTo>
```

```
</base:notifyAt>
```

```
<base:ExtensionGroup>
```

```
</base:ExtensionGroup>
```

```
<base:ExtensionAttributes/>
```

```
<base:status?</base:status>
```

The status of the job whose job request is being acknowledged

```
<base:statusDescription?</base:statusDescription>
```

```
<base:serviceProviderJobID?</base:serviceProviderJobID>
```

The identifier used by the server to identify this job, in its database

```
<base:queueReference>
```

```
<base:resourceID>0</base:resourceID>
```

```
<base:revisionID xsi:nil="true"/>
```

```
<base:location xsi:nil="true"/>
```

```
<base:resourceCreationDate xsi:nil="true"/>
```

```
<base:resourceModifiedDate xsi:nil="true"/>
```

```
<base:notifyAt>
```

```
<base:replyTo xsi:nil="true"/>
```

```
<base:faultTo xsi:nil="true"/>
```

```
</base:notifyAt>
```

```
< base:ExtensionGroup xsi:nil="true"/>
```

```
<base:ExtensionAttributes xsi:nil="true"/>
```

```
<base:status>started</base:status>
```

The identifier used by the server to identify the queue that the job is in

```
<base:statusDescription xsi:nil="true"/>
```

Queue status information

```
<base:length>4</base:length>
<base:availability>true</base:availability>
<base:estimatedTotalCompletionDuration xsi:nil="true"/>
<base:jobs xsi:nil="true"/>
</base:queueReference>
<base:tasks></base:tasks>
<base:operationName>?</base:operationName>
<base:bmObjects>
  <base:bmObject>
    <base:resourceID>?</base:resourceID>
    <base:revisionID>?</base:revisionID>
    <base:location>?</base:location>
    <base:resourceCreationDate>?</base:resourceCreationDate>
    <base:resourceModifiedDate>?</base:resourceModifiedDate>
    <base:notifyAt>
      <base:replyTo>?</base:replyTo>
      <base:faultTo>?</base:faultTo>
    </base:notifyAt>
    <base:ExtensionGroup>
    </base:ExtensionGroup>
    <base:ExtensionAttributes/>
  <base:bmContents>
    <base:bmContent>
      <base:resourceID>?</base:resourceID>
      <base:revisionID>?</base:revisionID>
      <base:location>?</base:location>
      <base:resourceCreationDate>?</base:resourceCreationDate>
      <base:resourceModifiedDate>?</base:resourceModifiedDate>
      <base:notifyAt>
        <base:replyTo>?</base:replyTo>
        <base:faultTo>?</base:faultTo>
      </base:notifyAt>
      <base:ExtensionGroup>
      </base:ExtensionGroup>
      <base:ExtensionAttributes/>
    <base:bmContentFormats>
      <base:bmContentFormat>
        <base:resourceID>?</base:resourceID>
        <base:revisionID>?</base:revisionID>
```

```

<base:location>?</base:location>
<base:resourceCreationDate>?</base:resourceCreationDate>
<base:resourceModifiedDate>?</base:resourceModifiedDate>
<base:notifyAt>
  <base:replyTo>?</base:replyTo>
  <base:faultTo>?</base:faultTo>
</base:notifyAt>
<base:ExtensionGroup>
  <Profile>
    <location>?</location>
    <name>?</name>
    <description>?</description>
    <destination>?<destination>
    <outputPattern>?< outputPattern >
    <outputFilePath>?< outputFilePath >
  </Profile>
</base:ExtensionGroup>
<base:ExtensionAttributes/>
<base:bmEssenceLocators>
  <base:bmEssenceLocator>
    <base:resourceID>?</base:resourceID>
    <base:revisionID>?</base:revisionID>
    <base:location>?</base:location>
    <base:resourceCreationDate>?</base:resourceCreationDate>
    <base:resourceModifiedDate>?</base:resourceModifiedDate>
    <base:notifyAt>
      <base:replyTo>?</base:replyTo>
      <base:faultTo>?</base:faultTo>
    </base:notifyAt>
    <base:ExtensionGroup>
    </base:ExtensionGroup>
    <base:ExtensionAttributes/>
    <base:storageType typeLabel="?" typeDefinition="?" typeLink="?">
      ?</base:storageType>
    <base:locatorInfo>?</base:locatorInfo>
    <base:containerMimeType typeLabel="?" typeDefinition="?" typeLink="?">
      ?</base:containerMimeType>
  </base:bmEssenceLocator>
</base:bmEssenceLocators>

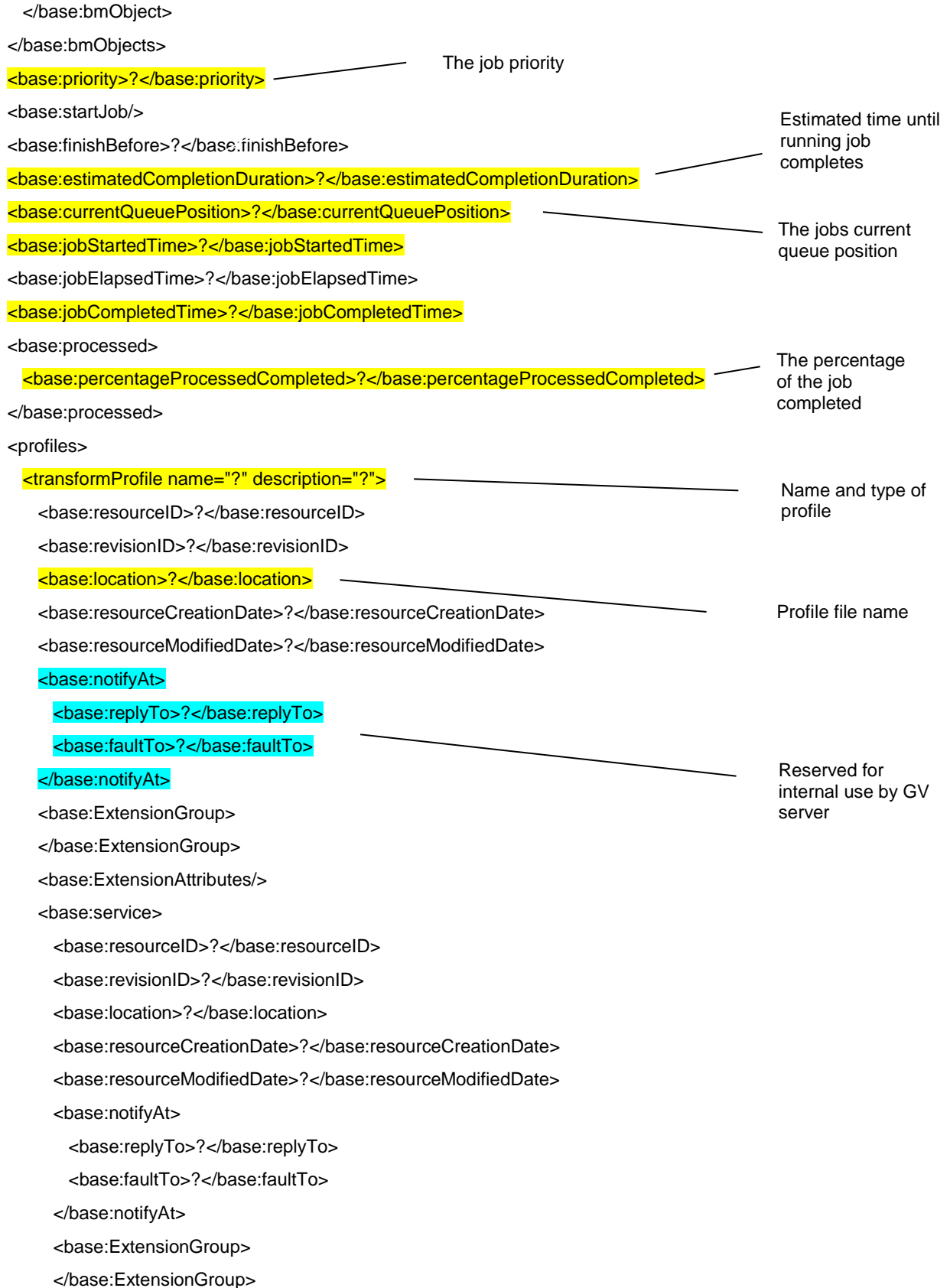
```

GV extension
Location = profile location
Name = profile name
Description = profile description
Destination = output folder
outputPattern = output file pattern
outputFilePath = actual output name

See Note 4 above for more details

The source file name?

```
<base:formatCollection>
  <!--Additional elements here-->
</base:formatCollection>
<base:duration>
  <base:timecode?></base:timecode>
  <base:normalPlayTime?></base:normalPlayTime>
  <base:editUnitNumber editRate="?" factorNumerator="?" factorDenominator="?">
    ?</base:editUnitNumber>
</base:duration>
<base:hash>
  <base:hashFunction typeLabel="?" typeDefinition="?" typeLink="?">
    ?</base:hashFunction>
  <base:value>cid:718883139875</base:value>
</base:hash>
<base:packageSize?></base:packageSize>
<base:mimeType typeLabel="?" typeDefinition="?" typeLink="?"></base:mimeType>
</base:bmContentFormat>
</base:bmContentFormats>
<base:descriptions>
  <base:description>
    <base:resourceID?></base:resourceID>
    <base:revisionID?></base:revisionID>
    <base:location?></base:location>
    <base:resourceCreationDate?></base:resourceCreationDate>
    <base:resourceModifiedDate?></base:resourceModifiedDate>
    <base:notifyAt>
      <base:replyTo?></base:replyTo>
      <base:faultTo?></base:faultTo>
    </base:notifyAt>
    <base:ExtensionGroup>
    </base:ExtensionGroup>
    <base:ExtensionAttributes/>
    <des:bmContentDescription xml:lang="?">
      <!--Additional elements here-->
    </des:bmContentDescription>
  </base:description>
</base:descriptions>
</base:bmContent>
</base:bmContents>
```



```
<base:ExtensionAttributes/>
<base:providerName?></base:providerName>
<base:providerEndPoint?></base:providerEndPoint>
<base:serviceDescription>
</base:serviceDescription>
</base:service>
<transformAtom>
  <base:videoFormat></base:videoFormat>
  <base:audioFormat></base:audioFormat>
  <base:containerFormat></base:containerFormat>
  <base:ExtensionGroup>
</base:ExtensionGroup>
  <base:ExtensionAttributes/>
</transformAtom>
<transferAtom>
  <base:destination?></base:destination>
  <base:ExtensionGroup>
</base:ExtensionGroup>
  <base:ExtensionAttributes/>
</transferAtom>
  <outputFileNamePattern?></outputFileNamePattern>
</transformProfile>
</profiles>
</transformJob>
</tran:transformAck>
</soapenv:Body>
</soapenv:Envelope>
```

Destination folder name

Destination output filename pattern.

See Table 2 in the User Guide to for patterns available