FaxOut Service
Semantic Model and Service Interface

Status: Approved

Abstract: Network print devices have evolved to support additional multifunction services, in particular FaxOutService. When FaxOut Devices are installed in local office or enterprise networks, they need remote service, device, and job management capabilities so that administrators, operators, and end users can monitor their health and status. In addition, such FaxOut Devices need remote job submission capabilities so that operators and end users can create FaxOut Jobs without depending entirely on local console interfaces. This document defines a semantic model for service, device, and job management and job submission for these FaxOut Devices.

This document is a PWG Candidate Standard. For a definition of "PWG Candidate Standard ", see:

This document is available electronically at:
Copyright (C) 2011, The Printer Working Group. All rights reserved.

This document may be copied and furnished to others, and derivative works that comment on, or otherwise explain it or assist in its implementation may be prepared, copied, published and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice, this paragraph and the title of the Document as referenced below are included on all such copies and derivative works. However, this document itself may not be modified in any way, such as by removing the copyright notice or references to the Printer Working Group, a program of the IEEE-ISTO.

Title: FaxOut Service Semantic Model and Service Interface

The IEEE-ISTO and the Printer Working Group DISCLAIM ANY AND ALL WARRANTIES, WHETHER EXPRESS OR IMPLIED INCLUDING (WITHOUT LIMITATION) ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

The Printer Working Group, a program of the IEEE-ISTO, reserves the right to make changes to the document without further notice. The document may be updated, replaced or made obsolete by other documents at any time.

The IEEE-ISTO and the Printer Working Group, a program of the IEEE-ISTO take no position regarding the validity or scope of any intellectual property or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; neither does it represent that it has made any effort to identify any such rights.

The IEEE-ISTO and the Printer Working Group, a program of the IEEE-ISTO invite any interested party to bring to its attention any copyrights, patents, or patent applications, or other proprietary rights, which may cover technology that may be required to implement the contents of this document. The IEEE-ISTO and its programs shall not be responsible for identifying patents for which a license may be required by a document and/or IEEE-ISTO Industry Group Standard or for conducting inquiries into the legal validity or scope of those patents that are brought to its attention. Inquiries may be submitted to the IEEE-ISTO by e-mail at:

info@ieee-isto.org

The Printer Working Group acknowledges that the IEEE-ISTO (acting itself or through its designees) is, and shall at all times, be the sole entity that may authorize the use of certification marks, trademarks, or other special designations to indicate compliance with these materials.

Use of this document is wholly voluntary. The existence of this document does not imply that there are no other ways to produce, test, measure, purchase, market, or provide other goods and services related to its scope.
About the IEEE-ISTO

The IEEE-ISTO is a not-for-profit corporation offering industry groups an innovative and flexible operational forum and support services. The IEEE Industry Standards and Technology Organization member organizations include printer manufacturers, print server developers, operating system providers, network operating systems providers, network connectivity vendors, and print management application developers. The IEEE-ISTO provides a forum not only to develop standards, but also to facilitate activities that support the implementation and acceptance of standards in the marketplace. The organization is affiliated with the IEEE [http://www.ieee.org/] and the IEEE Standards Association [http://standards.ieee.org/].

For additional information regarding the IEEE-ISTO and its industry programs visit: [http://www.ieee-isto.org].

About the Printer Working Group

The Printer Working Group (or PWG) is a Program of the IEEE-ISTO. All references to the PWG in this document implicitly mean “The Printer Working Group, a Program of the IEEE ISTO.” The PWG is chartered to make printers and the applications and operating systems supporting them work together better. In order to meet this objective, the PWG will document the results of their work as open standards that define print related protocols, interfaces, data models, procedures and conventions. Printer manufacturers and vendors of printer related software would benefit from the interoperability provided by voluntary conformance to these standards.

In general, a PWG standard is a specification that is stable, well understood, and is technically competent, has multiple, independent and interoperable implementations with substantial operational experience, and enjoys significant public support.

Contact information:

The Printer Working Group
c/o The IEEE Industry Standards and Technology Organization
445 Hoes Lane
Piscataway, NJ 08854
USA

MFD Web Page: [http://www.pwg.org/mfd] MFD Mailing List: [mfd@pwg.org]

Instructions for subscribing to the MFD mailing list can be found at the following link: [http://www.pwg.org/mailhelp.html]

Members of the PWG and interested parties are encouraged to join the PWG and MFD WG mailing lists in order to participate in discussions, clarifications and review of the WG product.
1 Introduction

This document specifies the PWG abstract model for a FaxOutService of a Multifunction Device (MFD). Included in this document is the service specific terminology, data model, the theory of operation, the FaxOutService interfaces and the conformance requirements. The MFD FaxOutService abstract model include the functional model and interfaces of a FaxOutService.

2 Overview

The MFD Facsimile service addressed in this specification is the FaxOutService. The FaxOutService responds to queries about its capabilities, configuration and descriptive information. It responds to queries for information about the FaxOut Jobs and their associated Documents. It manages and processes FaxOut Jobs with their associated FaxOutJobTicket and stores the digital output. A FaxOut Client application contains a FaxOut Client. A FaxOut client application interacts with the end user to obtain the end user’s FaxOut Intent and uses the FaxOut Client to communicate with the FaxOutService that will satisfy the end user’s FaxOut Intent.

FaxOut Templates contain instructions representing preconfigured FaxOut intent that can be used as is or modified by the end user. Once the end user is satisfied with the FaxOut Template the FaxOut client application passes the FaxOut Job Template to the FaxOut Job Client for submission to the FaxOutService. FaxOut Templates may be obtained in a number of ways all of which are outside the scope of this specification.

The Faxing scenarios addressed in this specification range from walk-up users that use an MFD’s front panel to send FaxOut Jobs to remote users that use their computers to send FaxOut Jobs. Users may also use workflow applications in an enterprise to send FaxOut Jobs. For batch FaxOut Jobs of single or multiple documents, the model supports automated sending of a stack of documents each separated by an individual FaxOut Instruction Sheet. A FaxOut Instruction Sheet is an implementation specific hard copy version of a FaxOutJobTicket and is outside the scope of this specification. The model also supports external security services that protect against unauthorized use of a FaxOutService and access of FaxOut digital data.
3 Terminology

3.1 Conformance Terminology

See [RFC2119] for conformance terminology used. There are no FaxOut-specific conformance terms.

3.2 Service Specific Terminology

See MFD Model and Common Semantics specification [PWG5108.01] for common MFD terminology used. For this service the "<service>" in the MFD Terminology section is replaced with "FaxOut". There is no FaxOutService specific terminology.

3.3 Model Mapping Conventions

The FaxOutService model is described in this document as an XML schema. This is for the sake of convenience and does not require a protocol mapping involving XML. The top level objects such as the Subunits, the Services, and their associated Jobs and Documents can be represented in any number of ways. In the abstract they are objects which contain attributes or properties that express characteristics of the object. Within this document references to Attribute or Element refer to XML Attributes and XML Elements respectively. Either of these can be abstractly considered to be attributes or properties of abstract objects.

3.4 Naming Conventions

The MFD Model and Common Semantics specification [PWG5108.01] describes common concepts and terms used for all of the services hosted on a Multifunction Device. Also included in the specification are the definitions of the objects and their attributes in the Multifunction Device data model. The MFD Model and Common Semantics specification [PWG5108.01] uses abstract names for the semantic elements (e.g., Job State). This document describes a specific service and uses an XML schema to represent the objects and attributes. XML elements cannot have names with an embedded whitespace. The names for objects and their attributes used by this specification are the names from the XML Schema (e.g., JobState). The names can be easily mapped between the two specifications by inserting or removing the whitespace in the name (e.g., Job State ≡ JobState).

4 Requirements

4.1 Rationale for the FaxOutService Specification

This specification is based on common requirements defined in the Multifunction Device Service Model Requirements [MFDREQ]. In order to support common functionality for faxing using Multifunction Devices, there is a clear need to develop a semantic model and a set of abstract operations and elements for FaxOut related services. In order to implement an abstract model of the operations and elements for FaxOut related services, there is need to map them onto implementable applications and communication protocols that support interactions between FaxOut Clients and FaxOutServices. There is a clear need to define a binding of the abstract model into Web Service Schema and Web Service protocol stack. As with other MFD Service specifications (e.g., Print Service, Scan Service, Copy Service) the Web Service binding requires a set of WSDL and XML Schema files based on the associated specification.

4.2 Out of Scope for FaxOutService

The basic FaxOutService model defined in this document is targeted to support enterprise FaxOut applications. However this document does not specify any application specific semantics. The following are out of scope:

1. Semantics of any compound service that creates separate Jobs for each transform service or destination URI specified in the original Job.
2. Semantics of any workflow protocol, i.e., sequencing and coordination of FaxOut jobs across multiple services.
3. Semantics for the creation of new document or file formats.

5 MFD Model Overview

See [PWG5108.01] for the MFD model. The FaxOutService fits within the MFD model as one of a number of services that can be hosted on a Multifunction Device (i.e., System). The critical MFD container Group Element with regard to describing the FaxOutService is Services.

One of the MFD's services is the FaxOutService. There can be multiple instances of a FaxOutService hosted on a Multifunction Device. This allows an implementation to expose multiple queues each with its own set of defaults and capabilities.

The System has a SystemConfiguration Group Element that contains all the subunits that comprise the MFD. Each FaxOutService instance contains a Service-specific view of the subunits used by that service instance. The FaxOutService element FaxOutServiceConfiguration contains the Service-specific view of the associated Subunits.

6 FaxOutService Model Overview

Figure 1 is the top level view of the FaxOutService schema.
The PWG semantic model supports zero or more FaxOutServices. A FaxOutService is hosted locally on an MFD or remotely on another computer. The FaxOutService model has an Active Job queue, a Job History and a set of Elements which includes FaxOutService status, configuration, description, defaults, and processing capabilities.

The FaxOutServiceCapabilities Group Element represents the allowed values supported by the FaxOutService for a FaxOutJobTicket and FaxOutDocumentTicket. The details of each FaxOutServiceCapabilities Elements are specified in §6.2.

The FaxOutServiceCapabilitiesReady Group Element represents the allowed values for a FaxOutJobTicket/FaxOutDocumentTicket that do not require operator intervention (e.g., the media that is actually loaded in an input tray). The details are specified in §6.3.

The FaxOutService Configuration provides a FaxOutService-specific view into the Subunits that are associated with this service instance. Only Subunits that are used by the FaxOutService will appear in this element. The details of each subunit are detailed in §6.4. The System element provides an all-encompassing view of all the Subunits of the MFD.

The FaxOutServiceDefaults Group Element contains FaxOutJobTicket and FaxOutDocumentTicket default values. The values contained in the default tickets are the values that that will be used by the FaxOutService when processing a FaxOutJobTicket/FaxOutDocumentTicket which does not explicitly specify a different value. The values for this are populated in an implementation specific manner. The details of the DefaultFaxOutTicket are specified in §6.1.

The FaxOutServiceDescription Group Element includes descriptive information such as service name and information, and has an extension point for vendor specific information. These Description Elements are can be set by Administrators. Similar to FaxOutService state elements, there are localized Description Elements for each supported Description Element. The details of the FaxOutServiceDescription Elements are specified in §6.5.

The FaxOutServiceStatus Group Element is an extension of the SystemServiceStatus class that includes elements such as ID, state, service counters, state messages and state reasons. State messages are localized state reasons. The only FaxOutService-specific status extensions are the FaxOutService-specific counters. The details of the Elements in the FaxOutServiceStatus group are specified in §6.6.

A FaxOutService contains zero or more jobs. Each job has a zero or more Documents which reference a Destination where the Digital Document(s) are stored as files. The FaxOutService organizes its FaxOutJobs in two job queues: (1) ActiveJobs, (2) JobHistory. ActiveJobs is a queue maintaining a list of jobs that are pending or processing. The JobHistory queue maintains a log of FaxOut that have reached a terminating state (i.e., Completed, Aborted, or Canceled). The retention period for jobs in the JobHistory list is implementation specific but MUST NOT be less than 300 seconds. Before a Job can be deleted from the JobHistory, the Job metadata (e.g., JobId, DateTimeAtCompleted, JobOriginatingUserName, DestinationUris) MUST be durably logged.

Each FaxOutJob can contain a FaxOutJobTicket which provides descriptive information as well as JobProcessing and DocumentProcessing instructions. The DocumentProcessing instructions apply to all documents within the job unless overridden at the document level with a FaxOutDocumentTicket.

Each FaxOutJob contains zero or more FaxOutDocuments. There is a time between the creation of a job and when the first document is added that the number of documents is zero. Support of multidocument jobs is OPTIONAL and implementation specific. The service’s support for multidocument jobs can be determined by examining the MultipleDocumentJobsSupported element in FaxOutServiceDescription. Note that PSTN FaxOut jobs either: (a) do not support multiple documents; or (b) do not distinguish the document boundaries in the FaxOut transmission.

### 6.1 FaxOutServiceDefaults

The FaxOutServiceDefaults provides the values that will be used if the element is omitted in a FaxOutJob’s FaxOutJobTicket or FaxOutDocumentTicket. Note that the processing instructions are not bound to the FaxOutJob until the FaxOutJob is actually processed. The values from the FaxOutServiceDefaults MUST NOT be copied to the Job’s FaxOutJobTicket or the Document’s FaxOutDocumentTicket. If the FaxOutJobReceipt is supported, the
combined elements from the user supplied FaxOutJobTicket and the applied values from the
DefaultFaxOutJobTicket are copied to the FaxOutJobReceipt. The DefaultFaxOutJobTicket elements MUST NOT
be copied to the Job's FaxOutJobTicket or the Document's FaxOutDocumentTicket. Similarly if the
FaxOutDocumentReceipt is supported, the combined elements from the user supplied FaxOutDocumentTicket and
the applied values from the DefaultFaxOutDocumentTicket are copied to the FaxOutDocumentReceipt.

Figure 2 FaxOutServiceDefaults

For descriptions of the elements that comprise FaxOutDocumentDescription, FaxOutJobDescription,
FaxOutJobProcessing and FaxOutDocumentProcessing see §7.3 on FaxOutJobTicket

6.2 FaxOutServiceCapabilities

The FaxOutServiceCapabilities provides information about the elements that can be used in FaxOutJobTickets and
FaxOutDocumentTickets. The values of the elements in FaxOutServiceCapabilities indicate all the supported
values for a FaxOutJobTicket and FaxOutDocumentTicket submitted to the FaxOutServer instance (e.g., all the
available resolutions, the available job priorities). The names of the elements within the FaxOutServiceCapabilities
are the same as those in the FaxOutJobTicket. See §7.3 and §8.3 for the names of the
FaxOutDocumentDescriptionCapabilities, FaxOutDocumentProcessingCapabilities,
FaxOutJobDescriptionCapabilities, and FaxOutJobProcessingCapabilities elements.

Although most of the elements have the same name as their FaxOutJobTicket counterparts the syntax is often
different. For example an element such as InputSource that is a single keyword in FaxOutJobTicket is a sequence
of keywords in FaxOutServiceCapabilities. The values list the allowed values for the FaxOutJobTicket element.
Some elements that are of the data type integer in a FaxOutJobTicket are a range of integers in
FaxOutServiceCapabilities. Other elements that are simple strings or predefined ranges in the FaxOutJobTicket
are simply Boolean values in FaxOutServiceCapabilities, indicating supported FaxOutJobTicket elements. See
[PWG5108.01] for details on the syntax.
Figure 3 FaxOutServiceCapabilities

Table 1 FaxOutServiceCapabilities

<table>
<thead>
<tr>
<th>Group</th>
<th>Elements described in [PWG5108.01]</th>
</tr>
</thead>
<tbody>
<tr>
<td>FaxOutDocumentProcessingCapabilities</td>
<td>NumberUp, PresentationDirectionNumberUp, AutoSkewCorrection, ContentType, Exposure, HeaderPrint, Quality, Resolutions, Scaling, DocumentSizeAutoDetect, ScanRegions, sides</td>
</tr>
<tr>
<td>FaxOutJobDescriptionCapabilities</td>
<td>ElementsNaturalLanguage, JobAccountingID, JobAccountingUserID, JobMandatoryElements, JobMessageFromOperator, JobMessageToOperator, JobMorInfo, JobName, JobOriginatingUserName, JobOriginatingUserUri, JobPassword, JobPasswordEncryption, KOctets, TemplateCreatorName, TemplateId, TemplateInfo, TemplateName, TemplateType, CompressionSupplied, DocumentCharsetSupplied, DocumentDigitalSignatureSupplied, DocumentFormatDetailsSupplied, DocumentFormatSupplied, DocumentFormatVersionSupported, DocumentMessageSupplied, DocumentNameSupplied</td>
</tr>
<tr>
<td>FaxOutJobProcessingCapabilities</td>
<td>JobDelayOutputUntil, JobDelayOutputUntilTime, JobHoldUntil, JobHoldUntilTime, JobMandatoryElements, JobPhoneNumber,</td>
</tr>
</tbody>
</table>

Copyright © 2011 The Printer Working Group. All rights reserved.
Resolution in the FaxOutJobTicket’s FaxOutDocumentProcessing is a single instance of Resolution from
the sequence of Resolutions element in FaxOutDocumentProcessingCapabilities.

For the FaxOut Service holding or delaying a job means to delay the transmission of the facsimile.

### 6.3 FaxOutServiceCapabilitiesReady

FaxOutServiceCapabilitiesReady provides information about the elements that can be used in FaxOutJobTickets
and FaxOutDocumentTickets. The values of the elements in FaxOutServiceCapabilitiesReady indicate all the
values for a FaxOutJobTicket and FaxOutDocumentTickets that can be submitted to the FaxOutServer instance
and applied without operator intervention. The names of the elements within FaxOutServiceCapabilitiesReady are
the same as those in FaxOutServiceCapabilities. See §7.3 and §8.3 for the names of the
FaxOutDocumentDescriptionCapabilities, FaxOutDocumentProcessingCapabilities,
FaxOutJobDescriptionCapabilities, and FaxOutJobProcessingCapabilities elements.

### 6.4 FaxOutServiceConfiguration

The types of Subunits defined in the MFD Model and Common Semantics specification [PWG5108.01] that are
applicable to a FaxOutService are Console, Cover, FaxModem, Finisher, InputChannel, InputTray, Interface,
Interpreter, Marker, MediaPath, OutputChannel, OutputTray, Processor, ScanMediaPath, Scanner, Storage and
optionally VendorSubunits. There are no standard subunits unique to the FaxOutService.
Figure 4 FaxOutServiceConfiguration

6.5 FaxOutServiceDescription

Figure 5 is a view of the Description elements for the FaxOutService. The Description elements provide descriptive information for the FaxOutService. The elements are administratively set. The element values can be modified directly or modified indirectly through an operation.
FaxOut Service Semantic Model and Service Interface

August 9, 2011

Copyright © 2011 The Printer Working Group. All rights reserved.
The elements common to all Service Descriptions are also described in the MFD Model and Common Semantics specification [PWG5108.01]. Those elements are identified in Figure 5 by being included in the yellow box. The remaining elements are common elements used by FaxOut. These elements are also described in the MFD Model and Common Semantics specification [PWG5108.01]. Note that FaxLogURI specifies the location of the durable log that is REQUIRED for all FaxOutService implementations (see section 7).

6.6 FaxOutServiceStatus

Figure 6 is a view of the Status elements for the FaxOutService. The Status elements provide state information for the FaxOutService. The elements are maintained by automata and cannot be directly set. The element values can be modified indirectly through an operation. For example PauseFaxOutService operation on the FaxOutService may result in the change of the State and StateReasons elements.
The elements common to all `<service>`ServiceStatus are described in the MFD Model and Common Semantics specification [PWG5108.01]. Those elements are identified in Figure 6 by being included in the yellow box. The remaining elements are common elements used by FaxOut. These elements are also described in the MFD Model and Common Semantics specification [PWG5108.01].
7 FaxOutJob Model

Figure 7 is the top level view of the FaxOutJob. The jobs appear in one of two lists. Pending and active jobs appear in the ActiveJobs list. Jobs that have reached a terminal state (i.e., Completed, Aborted, and Canceled) appear in the JobHistory list. Note that the JobHistory list is REQUIRED and the retention period for jobs in the JobHistory list is implementation specific but MUST NOT be less than 300 seconds. Before a Job can be deleted from the JobHistory, the Job metadata (e.g., JobId, DateTimeAtCompleted, JobOriginatingUserName, DestinationUris) MUST be durably logged to the location specified in FaxLogURI (see section 6.5).

FaxOutJobs can contain zero or more documents. During job creation it is possible that temporarily there are zero documents. The state of the job is described in the FaxOutJobStatus element. FaxOutJobTicket contains descriptive information about the job and the Job and Document processing instructions. It is possible to override the Document Processing instructions on a document by document basis by supplying a FaxOutDocumentTicket with the FaxOutDocument. The FaxOutJobTicket and FaxOutDocumentTicket represent the End User’s intent while the FaxOutJobReceipt and FaxOutDocumentReceipt represent what the FaxOutService actually did.
This element has exactly the same structure as the FaxOutJobTicket. For each processing element of a FaxOutJob, it records the actual value used by the FaxOutService for processing the FaxOutJob. It contains the elements supplied by the FaxOut Client and applied to the job, any element or values substitutions made by the FaxOutService and any default elements or values applied by the FaxOutService. See §7.3 for element descriptions.

7.2 FaxOutJobStatus

Figure 9 is a view of the Status elements for the FaxOutJob. The Status elements provide state information for the FaxOutJob. The elements are maintained by automata and cannot be directly set. The element values can be modified indirectly through an operation. For example CancelFaxOutJob operation on the FaxOutJob may result in the change of the State and StateReasons elements.
The elements common to all `<service>`JobStatus are described in the MFD Model and Common Semantics specification [PWG5108.01]. Those elements are identified in Figure 9 by being included in the yellow box. The remaining elements are common elements used by FaxOut. These elements are also described in the MFD Model and Common Semantics specification [PWG5108.01].
7.3 FaxOutJobTicket

The FaxOutJobTicket contains description and processing elements provided by the FaxOut Client during FaxOutJob creation. This information is used by the FaxOutService during the processing of a FaxOutJob. This information is made available to FaxOut Clients through the GetFaxOutJobElements operation and a subset is made available through the GetActiveFaxOutJobs and GetFaxOutJobHistory operations.

![Figure 10 FaxOutJobTicket](image)

7.3.1 FaxOutDocumentProcessing

The FaxOutDocumentProcessing provides the document processing instructions that have been requested by the End User at the job level. Each element has a MustHonor attribute to indicate whether documents within the job must be processed according to what the user has requested.
The elements common to all <service>DocumentProcessing are described in the MFD Model and Common Semantics specification [PWG5108.01]. Those elements are identified in Figure 11 by being included in the yellow box. The remaining elements are common elements used by FaxOut. These elements are also described in the MFD Model and Common Semantics specification [PWG5108.01].
7.3.2 FaxOutJobDescription

Figure 12 is a view of the Description elements for the FaxOutJob. These elements are set by the FaxOut Client during job creation.
The elements common to all Job Descriptions are described in the MFD Model and Common Semantics specification [PWG5108.01]. Those elements are identified in Figure 12 by being included in the yellow box. The remaining elements are common elements used by FaxOut. These elements are also described in the MFD Model and Common Semantics specification [PWG5108.01].

### 7.3.3 FaxOutJobProcessing

The FaxOutJobProcessing provides the job processing instructions that have been requested by the End User. Each element has a MustHonor attribute. When the value of MustHonor is true, the FaxOutService does not process the job unless the element is supported; otherwise the FaxOutService processes the job with its best effort.

![Diagram of FaxOutJobProcessing](image)

**Figure 13 FaxOutJobProcessing**

The elements common to all <service>JobProcessing are described in the MFD Model and Common Semantics specification [PWG5108.01]. Those elements are identified in Figure 13 by being included in the yellow box. The DestinationUrIs element is FaxOutService specific and is described in the next section. The remaining elements are common elements used by FaxOut. These elements are also described in the MFD Model and Common Semantics specification [PWG5108.01].
7.3.3.1 DestinationUris

This FaxOutJobProcessing element specifies destinations for the Fax transmission. It is a sequence of DestinationUrisEntry each of which defines a single destination. The DestinationUri is an RFC3966 [RFC3966] compliant URI. Dial Strings were dropped when RFC3966 superseded RFC2806 [RFC2806]. "Dial strings" are the actual numbers, symbols, and pauses entered by a user to place a phone call. We have added some elements to contain the Dial String information even though they could be encoded as RFC3966 compliant parameters. The PreDialString is the Dial string entered before the DestinationUri is applied. The PostDialString is the Dial string entered after the DestinationUri is applied.

The ABNF for a Dial Sting that applies to a PreDialString and a PostDialString is given below.

\[
\begin{align*}
\text{DialString} & = 1^* (\text{phonedigit} / \text{dtmf-digit} / \text{pause-character}) \\
\text{pause-character} & = \text{one-second-pause} / \text{wait-for-dial-tone} \\
\text{one-second-pause} & = "p" \\
\text{wait-for-dial-tone} & = "w" \\
\text{dtmf-digit} & = "*" / "A" / "B" / "C" / "D" / "#" \\
\text{phonedigit} & = \text{DIGIT} / [ \text{visual-separator} ] \\
\text{visual-separator} & = "." / ":" / ":" / ":" / ":" / ":" / ":" / ":"
\end{align*}
\]

The T33 subaddressing can be specified by the T33Subaddress element since a T33 subaddress can only contain digits [RFC3192], the datatype for the T33Subaddress is an integer. Multiple T33 subaddresses are not supported in a single DestinationUrisEntry. If multiple mailboxes are to be addressed each one requires its own entry.

8 FaxOutDocument Model

Figure 14 is the top level view of the FaxOutDocument. FaxOutDocuments are associated with one FaxOutJob. Note that PSTN FaxOut jobs either: (a) do not support multiple documents; or (b) do not distinguish the document boundaries in the FaxOut transmission. Initial implementations MAY support multiple document jobs. The state of the Document is described in the FaxOutDocumentStatus element. FaxOutJobTicket contains descriptive information about the job and the Job and Document processing instructions. It is possible to override the Document Processing instructions on a document by document basis by supplying a FaxOutDocumentTicket with the FaxOutDocument. The FaxOutJobTicket and FaxOutDocumentTicket represent the End User’s intent while the Job Receipt and Document Receipt represent what the FaxOutService actually did.
8.1 FaxOutDocumentReceipt

This element has exactly the same structure as the FaxOutDocumentTicket. For each processing element of a FaxOutDocument, it records the actual value used by the FaxOutService for processing the FaxOutDocument. It contains the elements supplied by the FaxOut Client, Any substitutions made by the FaxOutService and any Default elements applied by the FaxOutService. See §8.3 for element descriptions.

8.2 FaxOutDocumentStatus

Figure 15 is a view of the Status elements for the FaxOutDocument. The Status elements provide state information for the FaxOutDocument. The elements are maintained by automata and cannot be directly set.
Figure 15 FaxOutDocumentStatus
The elements common to all `<service>`DocumentStatus are described in the MFD Model and Common Semantics specification [PWG5108.01]. Those elements are identified in Figure 15 by being included in the yellow box. The remaining elements are common elements used by FaxOut. These elements are described in the MFD Model and Common Semantics specification [PWG5108.01]

### 8.3 FaxOutDocumentTicket

The FaxOutDocumentTicket contains description and processing elements provided by the FaxOut Client during FaxOutDocument creation. This information is used by the FaxOutService during the processing of a FaxOutDocument. This information is made available to FaxOut Clients through the GetFaxOutDocumentElements operation.

![Diagram of FaxOutDocumentTicket](image)

**Figure 16 FaxOutDocumentTicket**
8.3.1 FaxOutDocumentDescription

Figure 17 is a view of the Description elements for the FaxOutDocument. These elements are set by the FaxOut Client during Document creation.

The elements common to all <service>DocumentDescription are described in the MFD Model and Common Semantics specification [PWG5108.01]. Those elements are identified in Figure 17 by being included in the yellow box. The only other element included is the standard extension point.

8.3.2 FaxOutDocumentProcessing

This Group Element has exactly the same structure as the FaxOutDocumentProcessing element of FaxOutJob (See Error! Reference source not found.). It provides the document processing instructions that have been requested by the End User at each document level, overriding the job level document processing instructions.

9 FaxOutService Interfaces

The FaxOutService provides a set of service interfaces that is the same for a co-located local FaxOut Client or a remote FaxOut Client via a local interface, a local area network, or the Internet. A user makes a FaxOutService
request by interacting directly with the FaxOutService or indirectly through a local FaxOut Client via the MFD UI or a remote FaxOut Client via its software application UI.

The requests parameters are summarized in Table 2, Table 3 and Table 4. Responses to operations that fail are expected to return a fault.

The semantics for these operations are the same as the operations specified in the MFD Model and Common Semantics specification [PWG5108.01]. In Table 2, Table 3 and Table 4 the required parameters are in bold and optional parameters are in italic font.

---

### Table 2 Mandatory User Operations

<table>
<thead>
<tr>
<th>User Operation Name</th>
<th>Input Parameters</th>
<th>Output Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>AddFaxOutHardCopyDocument</td>
<td><strong>InputSource, JobId, ElementsNaturalLanguage, FaxOutDocumentTicket, LastDocument, RequestingUserName</strong></td>
<td><strong>DocumentNumber, UnsupportedAttributes</strong></td>
</tr>
<tr>
<td>CancelFaxOutJob</td>
<td><strong>ElementsNaturalLanguage, JobId, Message, RequestingUserName</strong></td>
<td></td>
</tr>
<tr>
<td>CloseFaxOutJob</td>
<td><strong>JobId, RequestingUserName</strong></td>
<td></td>
</tr>
<tr>
<td>CreateFaxOutJob</td>
<td><strong>ElementsNaturalLanguage, FaxOutJobTicket, RequestingUserName</strong></td>
<td></td>
</tr>
<tr>
<td>GetActiveFaxOutJobs</td>
<td><strong>ElementsNaturalLanguageRequested, limit, RequestingUserName</strong></td>
<td><strong>ActiveJobs(list of JobSummary), ElementsNaturalLanguage</strong></td>
</tr>
<tr>
<td>GetFaxOutJobElements</td>
<td><strong>ElementsNaturalLanguageRequested, JobId, RequestedElements, RequestingUserName</strong></td>
<td><strong>FaxOutJobElements, ElementsNaturalLanguage</strong></td>
</tr>
<tr>
<td>GetFaxOutJobHistory</td>
<td><strong>ElementsNaturalLanguageRequested, limit, RequestingUserName</strong></td>
<td><strong>JobHistory(list of JobSummary), ElementsNaturalLanguage</strong></td>
</tr>
<tr>
<td>GetFaxOutServiceElements</td>
<td><strong>ElementsNaturalLanguageRequested, RequestedElements, RequestingUserName</strong></td>
<td><strong>ElementsNaturalLanguage, FaxOutServiceElements</strong></td>
</tr>
<tr>
<td>SendFaxOutDocument</td>
<td><strong>ElementsNaturalLanguageRequested, FaxOutDocumentTicket, JobId, LastDocument, RequestingUserName, DocumentData</strong></td>
<td><strong>DocumentNumber, UnsupportedAttributes</strong></td>
</tr>
<tr>
<td>ValidateFaxOutJobTicket</td>
<td><strong>ElementsNaturalLanguage, FaxOutJobTicket, RequestingUserName</strong></td>
<td><strong>UnsupportedAttributes</strong></td>
</tr>
</tbody>
</table>

### Table 3 Optional User Operations

<table>
<thead>
<tr>
<th>User Operation Name</th>
<th>Input Parameters</th>
<th>Output Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>CancelCurrentFaxOutJob</td>
<td><strong>ElementsNaturalLanguage, JobId, Message, RequestingUserName</strong></td>
<td></td>
</tr>
<tr>
<td>CancelFaxOutDocuments</td>
<td><strong>DocumentNumber, ElementsNaturalLanguage, JobId, Message, RequestingUserName</strong></td>
<td></td>
</tr>
<tr>
<td>User Operation Name</td>
<td>Input Parameters</td>
<td>Output Parameters</td>
</tr>
<tr>
<td>-----------------------------</td>
<td>-----------------------------------------------------------------------------------</td>
<td>------------------------------</td>
</tr>
<tr>
<td>CancelMyFaxOutJob</td>
<td><code>ElementsNaturalLanguage, JobIds, Message, RequestingUserName</code></td>
<td><code>JobIds</code></td>
</tr>
<tr>
<td>GetFaxOutDocumentElements</td>
<td><code>DocumentNumber, ElementsNaturalLanguageRequested, JobId, RequestedElements, RequestingUserName</code></td>
<td><code>FaxOutDocumentElements, ElementsNaturalLanguage</code></td>
</tr>
<tr>
<td>GetFaxOutDocuments</td>
<td><code>ElementsNaturalLanguageRequested, JobId, RequestingUserName</code></td>
<td><code>Documents(List of DocumentSummary), ElementsNaturalLanguage, JobId, JobName</code></td>
</tr>
<tr>
<td>HoldFaxOutJob</td>
<td><code>ElementsNaturalLanguage, JobId, Message, RequestingUserName</code></td>
<td></td>
</tr>
<tr>
<td>ReleaseFaxOutJob</td>
<td><code>ElementsNaturalLanguage, JobId, Message, RequestingUserName</code></td>
<td></td>
</tr>
<tr>
<td>ResubmitFaxOutJob</td>
<td><code>ElementsNaturalLanguage, FaxOutJobTicket, JobId, RequestingUserName</code></td>
<td><code>JobId, UnsupportedElements</code></td>
</tr>
<tr>
<td>ResumeFaxOutJob</td>
<td><code>ElementsNaturalLanguage, JobId, Message, RequestingUserName</code></td>
<td></td>
</tr>
<tr>
<td>SendFaxOutUri</td>
<td><code>DocumentUri, ElementsNaturalLanguageRequested, FaxOutDocumentTicket, JobId, LastDocument, RequestingUserName</code></td>
<td><code>DocumentNumber, UnsupportedAttributes</code></td>
</tr>
<tr>
<td>SetFaxOutDocumentElements</td>
<td><code>DocumentNumber, ElementsNaturalLanguage, FaxOutDocumentTicket, JobId, Message, RequestingUserName</code></td>
<td><code>UnsupportedElements</code></td>
</tr>
<tr>
<td>SetFaxOutJobElements</td>
<td><code>ElementsNaturalLanguage, FaxOutJobTicket, JobId, Message, RequestingUserName</code></td>
<td><code>UnsupportedElements</code></td>
</tr>
<tr>
<td>SuspendCurrentFaxOutJob</td>
<td><code>ElementsNaturalLanguage, JobId, Message, RequestingUserName</code></td>
<td></td>
</tr>
<tr>
<td>ValidateFaxOutDocumentTicket</td>
<td><code>ElementsNaturalLanguage, FaxOutDocumentTicket, RequestingUserName</code></td>
<td><code>UnsupportedElements</code></td>
</tr>
</tbody>
</table>

Table 4 Administrative Operations

<table>
<thead>
<tr>
<th>Administrative Operation Name</th>
<th>Input Parameters</th>
<th>Output Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>CancelFaxOutJobs</td>
<td><code>ElementsNaturalLanguage, JobIds, Message, RequestingUserName</code></td>
<td><code>JobIds</code></td>
</tr>
<tr>
<td>DisableFaxOutService</td>
<td><code>ElementsNaturalLanguage, Message, RequestingUserName</code></td>
<td></td>
</tr>
<tr>
<td>EnableFaxOutService</td>
<td><code>ElementsNaturalLanguage, Message, RequestingUserName</code></td>
<td></td>
</tr>
<tr>
<td>HoldNewFaxOutJobs</td>
<td>`ElementsNaturalLanguage, JobHoldUntil</td>
<td>JobHoldUntilTime, Message, RequestingUserName`</td>
</tr>
</tbody>
</table>
### 10 Conformance Requirements

This section describes conformance issues and requirements. This document introduces model entities such as objects, operations, elements, element syntaxes, and element values. These conformance sections describe the conformance requirements which apply to these model entities.

#### 10.1 Client Conformance Requirements

A conforming FaxOut Client MUST support all REQUIRED operations as defined in this document. For each parameter included in an operation request, a conforming FaxOut Client MUST supply a value whose type and value syntax conforms to the requirements as specified in Section 5. A conforming FaxOut Client MAY supply any extensions in an operation request, as long as they meet the requirements in Section 10.4.

When sending a request, a conforming FaxOut Client MAY supply any parameters that are indicated as OPTIONALLY supplied by the FaxOut Client.

A FaxOut Client MUST be able to accept any of the elements defined in the model, including their full range that may be returned to it in a response from a FaxOutService.

An operation response can contain elements and/or values that the FaxOut Client does not expect. Therefore, a FaxOut Client implementation MUST gracefully handle such responses and not refuse to interoperate with a conforming FaxOutService that is returning extended elements and/or values that conform to Section 10.4. FaxOut Clients may choose to ignore any parameters, elements, or values that they do not understand.

#### 10.2 FaxOutService Conformance Requirements

This section specifies the conformance requirements for conforming implementations with respect to objects, operations, and attributes.

##### 10.2.1 Objects

Conforming implementations MUST implement all of the REQUIRED model objects as defined in this specification in the indicated sections:
Section 6 - FaxOutService

Section 7 - FaxOutJob

Section 8 - FaxOutDocument

10.2.2 Operations

Conforming FaxOutService implementations MUST implement all of the REQUIRED model operations, including REQUIRED responses, as defined in this specification in section 9:

- CreateFaxOutJob
- CloseFaxOutJob
- CancelFaxOutJob
- ValidateFaxOutJobTicket
- AddFaxOutHardCopyDocument
- GetActiveFaxOutJobs
- GetFaxOutJobHistory
- GetFaxOutServiceElements
- GetFaxOutJobElements
- SendFaxOutDocument

Conforming FaxOutService MUST support all REQUIRED operation elements and all values of such elements if so indicated in the description. Conforming FaxOutService MUST ignore all unsupported or unknown operation elements received in a request, but MUST reject a CreateFaxOutJob request that contains an unknown element that contains the MustHonor attribute with a value of ‘true’. Note that PSTN FaxOut jobs either: (a) do not support multiple documents; or (b) do not distinguish the document boundaries in the FaxOut transmission.

10.2.3 Job History

Conforming FaxOutService implementations MUST retain every Job in the JobHistory for at least 300 seconds (see section 7). Before a Job can be deleted from the JobHistory, the Job metadata (e.g., JobId, DateTimeAtCompleted, JobOriginatingUserName, DestinationUris) MUST be durably logged.

10.3 FaxOutService Elements

Conforming FaxOutService MUST support all of the REQUIRED object elements, as defined in this specification.

If an object supports an element, it MUST support only those values specified in this document or through the extension mechanism described in section 10.4. It MAY support any non-empty subset of these values. That is, it MUST support at least one of the specified values and at most all of them.

10.4 Extensions

Conforming FaxOutService MAY support extensions. To extend the model the extensions MUST be fully qualified. The qualified name MUST NOT be in the PWG target namespace. When extending the model with new elements the new elements MUST be added at the extension points at the end of the associated sequence of elements. Extended values for elements MUST conform to the extension patterns defined in the element schema. Implementers are free to add vendor specific operations to the service.

11 PWG and IANA Registration Considerations

This specification abides by the guidelines set forth in the MFD Model and Common Semantics specification [PWG5108.01] (section 10).

12 Internalization Considerations

This specification abides by the guidelines set forth in the MFD Model and Common Semantics specification [PWG5108.01] (section 11).
13 Security Considerations

13.1 Protection of End User’s FaxIn Data

This specification abides by the guidelines set forth in the MFD Model and Common Semantics specification [PWG5108.01] (section 12).

An end user’s FaxIn data can be protected from disclosure by encrypting the content and protected from modification by signing the data file when the data is stored in a repository or being transmitted over a communication link.

Signing or encrypting data files stored in a repository requires secure key management which includes the selection, generation, distribution, and destruction of effective signing or encryption of each end user’s keys. Signing or encrypting data files stored in a repository is outside the scope of the FaxInService. It is RECOMMENDED that the end user designates a repository that has their desired level of signing or encryption capabilities if so is required.

For protection of the data files transmitted over the network between a FaxInService and a repository, the FaxInService SHALL support the secure communication protocols required by the end user’s site policy, which may require signing and/or encryption of the transmitted document. The FaxInService SHALL have security attributes to indicate the signing or encryption support for the end user’s site policy. Different levels of security require different signing or encryption methods to be used. A site policy administrator SHOULD be responsible to manage the site security policy to ensure consistency with the site security requirements. The MFD implementer SHOULD ensure that a signing or encryption method consistent with the user’s site policy is used for transporting user’s FaxIn data over a shared communication medium.

14 References

14.1 Normative References


14.2 Informative References

[WS-FaxOut]  

[MFDREQ]  

[DES]

AES

ECC

MD5

MD4

MD2

SHA

CHMOD

RFC2806

STAT

15 Author's Address

Peter Zehler
Xerox Research Center Webster
Email: Peter.Zehler@Xerox.com
Voice: (585) 265-8755
FAX: (585) 265-7441
US Mail: Peter Zehler
Xerox Corp.
800 Phillips Rd.
M/S 128-25E
Webster NY, 14580-9701

Additional contributors:
Ira McDonald – High North
Glen Petrie – Epson
Lee Farrell
Bill Wagner – TIC