April 10, 2009
Candidate Standard 5108.02-2009

The Printer Working Group

Network Scan Service
Semantic Model and Service Interface

Version 1.0
Status: Approved

Abstract: Network print devices have evolved to support additional multifunction services, in particular Scan Service. When network Scanners 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 network Scanners need remote request for job creation capabilities so that operators and End Users can create Scan Jobs without depending entirely on local console interfaces. This document defines a semantic model for service, device, and job management and request for job creation for these network Scanners.

This document is a PWG Candidate Standard. For a definition of a "PWG Candidate Standard", see:
ftp://ftp.pwg.org/pub/pwg/general/pwg-process20.pdf
This document is available electronically at:
ftp://ftp.pwg.org/pub/pwg/candidates/cs-sm20-scan10-20090410-5108.02.pdf, .doc

Title: Network Scan Service Semantic Model and Service Interface

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.

CONTENTS

Figures

Tables

1 Introduction

This document specifies the PWG abstract model for scanning services of a Multifunction Device (MFD). Included in this document is the content specific terminology, data model, the theory of operation, the scanning service interfaces and the conformance requirements. The MFD scanning service abstract models include the functional models and interfaces of the associated scanning services for a local network or enterprise-connected multifunction device.

2 Summary

The MFD scanning service addressed in this specification is the Scan Service. The Scan Service responds to queries about its capabilities, configuration and descriptive information. It responds to queries for information about the Scan Jobs and their associated Documents. It manages and processes Scan Jobs with their associated ScanJobTicket and stores the digital output. A network scanning client application contains a Scan Client. A network scanning client application interacts with the End User to obtain the End User's Scan Intent and uses a Scan Client to communicate with the Scan Service that will execute the End User's Scan Intent.

Scan Templates contain instructions representing preconfigured Scan Intent that can be used as is or modified by the End User. Once the End User is satisfied with the Scan Template the network scanning client application passes the Scan Job Template to the Scan Job Client for submission to the Scan Service. Scan templates may be obtained in a number of ways. Those methods are outside the scope of this specification.

The scanning scenarios addressed in this specification ranges from walk-up scanning using the MFD's front panel to remote scanning from an End User's computer to support document on-ramp scanning using enterprise workflow applications. When shared by a workgroup using different functions of the MFD, the model supports interruption of a large Scan Job to perform other MFD functions including a different Scan Job. For batch job scanning of either single or multiple documents, the model supports automated scanning of a stack of documents separated by an individual Scan Instruction Sheet. The model also supports external security services that protect against unauthorized use of the scanning services and access of scanned digital data.

3 Terminology

3.1 Conformance Terminology

Capitalized terms, such as **MUST, MUST NOT, REQUIRED, SHOULD, SHOULD NOT, MAY, RECOMMENDED** and **OPTIONAL**, have special meaning relating to conformance as defined in RFC 2119 [RFC2119].

MUST This word means that the definition is an absolute requirement of the specification.

REQUIRED This word means that the definition is an absolute requirement of the specification.

SHALL This word means that the definition is an absolute requirement of the specification.

MUST NOT This phrase means that the definition is an absolute prohibition of the specification.

SHALL NOT This phrase means that the definition is an absolute prohibition of the specification.

SHOULD This word means that there may exist valid reasons in particular circumstances to ignore a particular item, but the full implications must be understood and carefully weighed before choosing a different course.

SHOULD NOT This phrase mean that there may exist valid reasons in particular circumstances when the particular behavior is acceptable or even useful, but the full implications should be understood and the case carefully weighed before implementing any behavior described with this label.

RECOMMENDED This word means that there may exist valid reasons in particular circumstances to ignore a particular item, but the full implications must be understood and carefully weighed before choosing a different course.

NOT RECOMMENDED This phrase mean that there may exist valid reasons in particular circumstances when the particular behavior is acceptable or even useful, but the full implications should be understood and the case carefully weighed before implementing any behavior described with this label.

MAY This word means that an item is truly optional. One vendor may choose to include the item because a particular marketplace requires it or because the vendor feels that it enhances the product while another vendor may omit the same item. An implementation which does not include a particular option MUST be prepared to interoperate with another implementation which does include the option, though perhaps with reduced functionality. In the same vein an implementation which does include a particular option MUST be prepared to interoperate with another implementation which does not include the option (except, of course, for the feature the option provides.)

OPTIONAL This word means that an item is truly optional. One vendor may choose to include the item because a particular marketplace requires it or because the vendor feels that it enhances the product while another vendor may omit the same item. An implementation which does not include a particular option MUST be prepared to interoperate with another implementation which does include the option, though perhaps with reduced functionality. In the same vein an implementation which does include a particular option MUST be prepared to interoperate with another implementation which does not include the option (except, of course, for the feature the option provides.)

3.2 Content Specific Terminology

Term	Definition
Active Jobs	Active Jobs are a Scan Service queue containing all the Scan Jobs that are waiting to be processed or are currently processing.
ADF	Automatic Document Feeder is a mechanism for handling Hard Copy Documents for scanning. The mechanism selects a media sheet from its input tray and passes it to the image acquisition subsystem of the Scanner. After the Scan is complete the ADF transports the Hard copy Document's media sheet to its final destination (e.g. output bin, ADF output bin)
Attribute	An attribute is an entity that defines a property of an object, element, or file. An attribute usually consists of a name and a value. Its definition can be extended by a datatype, a representation format, a default value and restrictions.
Cross Feed (direction)	Cross Feed is the direction perpendicular to the movement of the Hard Copy Document or light bar of the scanner. For scanners that use a technology other that a light bar, the Image data is acquired most quickly along this direction. The Cross Feed direction is also referred to as Fast Scan and X direction. (See §6.5.4)
Default Scan Job Ticket	The Default Scan Job Ticket is a Scan Job Ticket data object that is bound to a Scan Service. The attribute values contained in the Default Scan Job Ticket are the values that that will be used by the Scan Service when processing a Scan Job whose Scan Job Ticket does not specify a different value.
Destination	Destination is the location (i.e. URL) to store the Digital Document(s) from a Scan Job.
Destination URL	Destination URL is an alternative term for Destination(See above). (Note: This is represented in the model by the DestinationURI attribute)
Digital Document	A Digital Document is the output of a Scan Service containing the digitized data resulting from the scanning of a Hardcopy Document(s). The Images from the scanned Hardcopy Document(s) are encoded in an Image or document format and stored at a Destination.
Directory Service	A Directory Service is a software application or a set of applications that stores, organizes and provides access to information about a computer network's users and resources.
Discovery Client	A Discovery Client is a software application that performs service or resource discovery on a computer network.
Document Repository	A Document Repository is a local or remote data store where Digital Documents are stored by the Scan Service.
Element	Elements are used to convey structure and relationships in XML document instances. An Element can contain both content and Attributes.
Fast Scan (direction)	Fast Scan is the direction perpendicular to the movement of the Hard Copy Document or light bar of the scanner. For scanners that use a technology other that a light bar, the Image data is acquired most

Term	Definition
	quickly along this direction. The Fast Scan direction is also referred to as Cross Feed direction and X. (See §6.5.4)
Feed (direction)	Feed is the direction parallel to the movement of the Hard Copy Document or light bar of the scanner. For scanners that use a technology other that a light bar, the Image data is acquired most slowly along this direction. The Feed direction is also referred to as Slow Scan direction and Y. (See §6.5.4)
Group Element	A Group Element is a collection of Elements that constitutes a complex Element.
Hardcopy Document	AHardcopy Document is a physical document in the form of paper, transparency, film… etc. that is the input source for a Scan Job. The Hardcopy Document is scanned by the Scanner Subunit and the Images transformed by the Scan Service into a Digital Document and stored in at the Destination within a Document Repository.
Image	An Image is a digital (i.e. binary) representation of the information captured by a Scanner Subunit. One Image is produced as a result of a Scanner Subunit scanning a ScanRegion.
Job History	The Job History is a Scan Service queue containing all the Scan Jobs that have reached a terminating state. The terminating states are defined as Completed, Aborted and Canceled. The length of this queue is determined by the implementer. The Scan Jobs SHOULD remain in the Job History for a reasonable period of time to permit interested parties to obtain information on completed jobs.
Local Client	Local Client is an alternative term for Local Scan Client. (See Local Scan Client and Scan Client below.)
Local Scan Client	TheLocal Scan Client is a Scan Client application within the MFD. (See Scan Client below.)
Physical Scan Document Ticket	A Physical Scan Document Ticket is an encoded hardcopy Scan Document Ticket, directly marked by the End User, that becomes a Scan Document Ticket data object after being scanned and processed.
Physical Scan Job Ticket	A Physical Scan Job Ticket is an encoded hardcopy ScanJobTicket, directly marked by the End User, that becomes a ScanJobTicket data object after being scanned and processed.
Remote Scan Client	The Remote Scan Client is a Scan Client application external to the MFD. (See Scan Client below.)
Scan Client	The Scan Client is a local or remote software entity that interfaces with the End User and interacts with a Scan Service.
Scanner Subunit	TheScanning component of an MFD subsystem that is responsible for image acquisition and media handling (i.e. the scanner).
Scan Document	The Scan Document is a data object managed by a Scan Service that contains Document level description, processing, status information of a Document within a Scan Job.
Scan Document Data	Scan Document Data is a term interchangeable with Digital Document throughout this specification. (See Digital Document above)
Scan Document Ticket	The Scan Document Ticket is a data object that contains an End User's Scan Intent for document processing and descriptive properties of a Scan Document of a Scan Job. Any document

Term	Definition
	processing properties in the ScanDocumentTicket will override the values specified in the ScanJobTicket's document processing properties. The content of a ScanDocumentTicket is configured by End User through a Scan Client.
Scan Intent	Scan Intent is the End User's preferences for the processing and description properties of a Scan Job or Scan Document.
Scan Job	A Scan Job is a data object, created and managed by a Scan Service, that contains the description, processing, and status information of a job submitted by a user. The Scan Job can contain one or more Document objects. The purpose of a Scan Job is to acquire the digital content (Images) of scanned Hardcopy Documents and store them at a specified destination in a specified document format.
Scan Job Receipt	A Scan Job Receipt is an element of the Scan Service that contains information on the actual values of processing elements used by the Scan Service for processing a Scan Job. The content of a ScanJobReceipt is populated by the Scan Service when a Scan Job is processed.
Scan Region	A Scan Region is a rectangular region of the Scanner Subunit's Platen that has been specified by an Administrator or End User as the bounding area in which a scan will occur.
Scan Job Ticket	A Scan Job Ticket is a data object that contains an End User's Scan Intent for document processing, job processing and descriptive job properties of a Scan Job. The job elements apply to the entire Scan Job. The document processing elements will be used for all the Documents within the Scan Job unless overridden at the Document level (See ScanDocumentTicket). The content of a ScanJobTicket is configured by End User through a Scan Client.
Scan Job Template	A Scan Job Template is a Scan Job Ticket data object representing an End User's preconfigured Scan Intent that is not bound to a Scan Service or Scan Job.
Scan Service	The Scan Service is a software service that accepts and processes requests to create, monitor and manage Scan Jobs. The software service accepts and processes requests to monitor and control the status of the service itself and its associated resources. A Scan Service is hosted either locally or remotely to the MFD.
Sequence	A Sequence is an ordered list of elements.
Slow Scan (direction)	The Slow Scan direction is the direction along which the Image data is acquired most slowly. If the light bar or media moves, this direction is parallel to that movement The Slow Scan direction is also referred to as Feed direction and Y. (See §6.5.4)
X	X is an axis of the coordinate system. This axis is associated with the Fast Scan direction of the Scanner Subunit. If the light bar or Hard Copy Document moves, they do not move in the X direction. They move in the Y direction. This direction is also referred to as Cross Feed direction and Fast Scan direction. (See §6.5.4)
Y	Y is an axis of the coordinate system. This axis is associated with the slow scan direction of the Scanner Subunit. If the light bar or Hard Copy Document moves, they move in Y direction. This direction is also referred to as Feed direction and Slow Scan

Term	Definition
	direction. (See §6.5.4)

4 Rationale

4.1 Rationale for this Scanning Service Specification

In order to support common functionality for scanning using network multifunction devices, there is a clear need to develop a semantic model and a set of abstract operations and elements for scanning related services. In order to implement an abstract model of the operations and elements for scanning related services, there is need to map them onto implementable applications and communication protocols that support interactions between Scan Clients and Scan Services. There is a clear need to define a binding of the abstract model into Web Service Schema and Web Service protocol stack.

4.2 Out of Scope for Scan Service

The basic scanning service model defined in this document is targeted to support enterprise scan applications. However this document does not specify any application specific semantics. The MFD Working Group charter defines the following as out of scope:

1. Semantics of any compound service such as Scan-To-Email, Scan-To-Fax, Scan-To-Mailbox, or Scan-To-Print of which the additional semantics associated with accessing the specific document repositories will be defined in other services, not included in the scanning services.
2. Semantics of any workflow protocol, i.e., sequencing and coordination of scanning jobs across multiple services.
3. Semantics of any scanning service management operations for MFDs that are not network connected.
4. Semantics for the creation of new document or file formats.

4.3 Model Mapping Conventions

The Scan Service 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. Abstractly they are objects which contain attributes or properties that express characteristics of the object. For the remainder of 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.

5 MFD Model Overview

Below is the top level view of the Multifunction Device model and the Scan Services place within it.

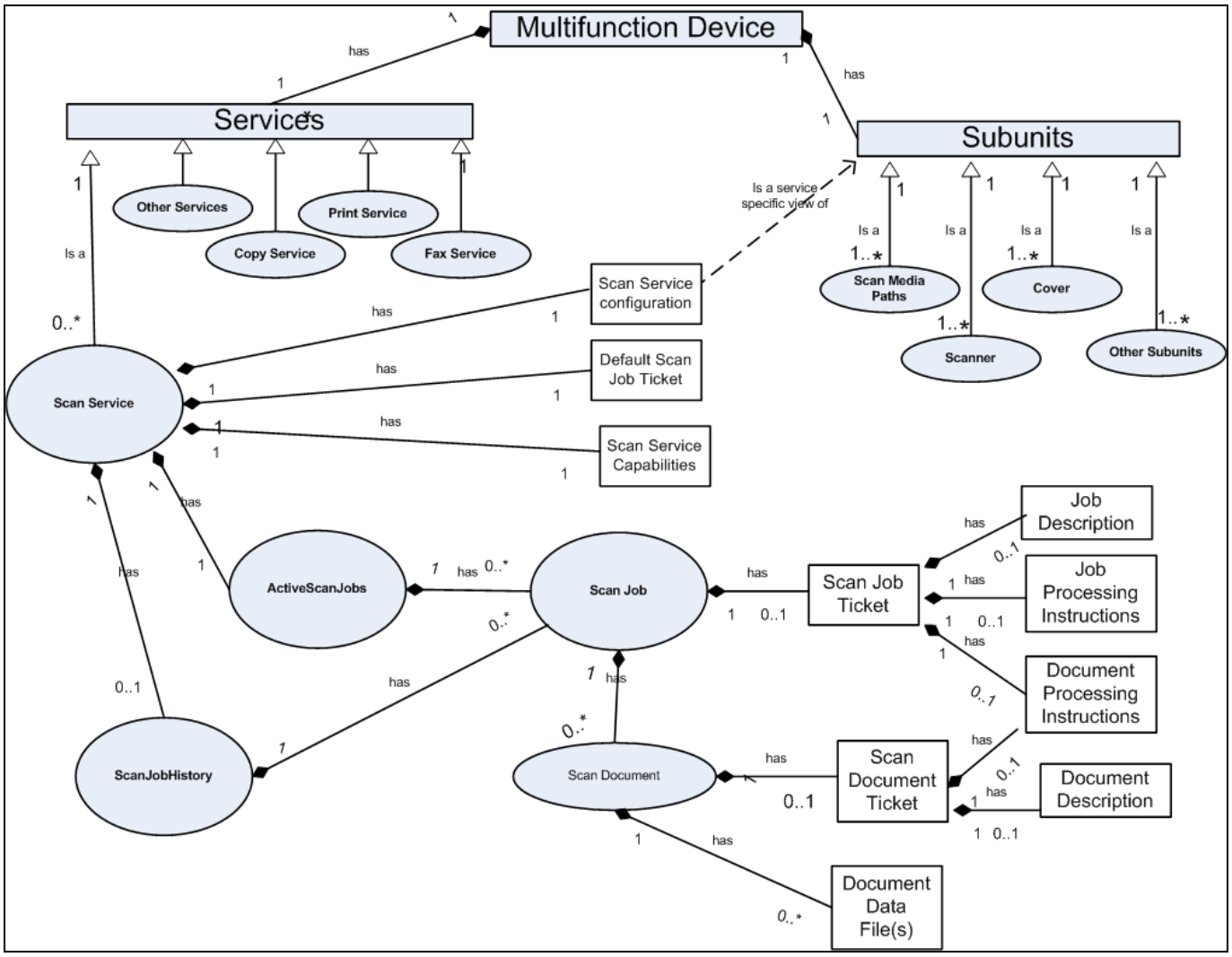


Figure 1 Multifunction Device Overview

The Scan Service is just one of a number of services that can be hosted on a multifunction device. The multifunction device is represented as the Server element in the PWG Semantic Model Schema. The Multifunction Device contains a number of objects critical to describing the scan related functionality such as Scan Service and Subunits.

The Multifunction Device contains all the Services and all the Subunits that comprise the Multifunction Device. One of the services is the Scan Service. There can be multiple instances of a Scan Service hosted on a Multifunction Device. This allows an implementation to expose multiple queues each with its own set of defaults and capabilities.

Each Scan Service instance contains a ScanServiceConfiguration which contains only the Subunits used by that service instance. Each Service also contains a DefaultScanJobTicket and ScanServiceCapabilities. These provide the Service's default ScanJobTicket values and the allowed values for the ScanJobTicket respectively.

Each Scan Service contains zero or more Scan Jobs. The Scan Jobs appear either in the ActiveJobs list or in the JobHistory list. Jobs waiting to be processed or currently processing are on the ActiveJobs list. Jobs that have reached a terminating state (i.e. Completed, Aborted, and Canceled) are on the JobHistory list. Whether or not the JobHistory list is implemented or how long jobs remain on the JobHistory list is implementation specific.

Each Scan Job contains zero or more ScanDocuments. There is a period of 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 implementation specific. The service's support for multidocument jobs can be determined by examining the Scan Service Capabilities. Each Scan Job can contain a ScanJobTicket 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 ScanDocumentTicket. The relationship between Scan Jobs, ScanDocuments and Document Data Files is given below in section 6.2.

6 Scan Service Model Overview

Below is the top level view of the Scan Service schema.

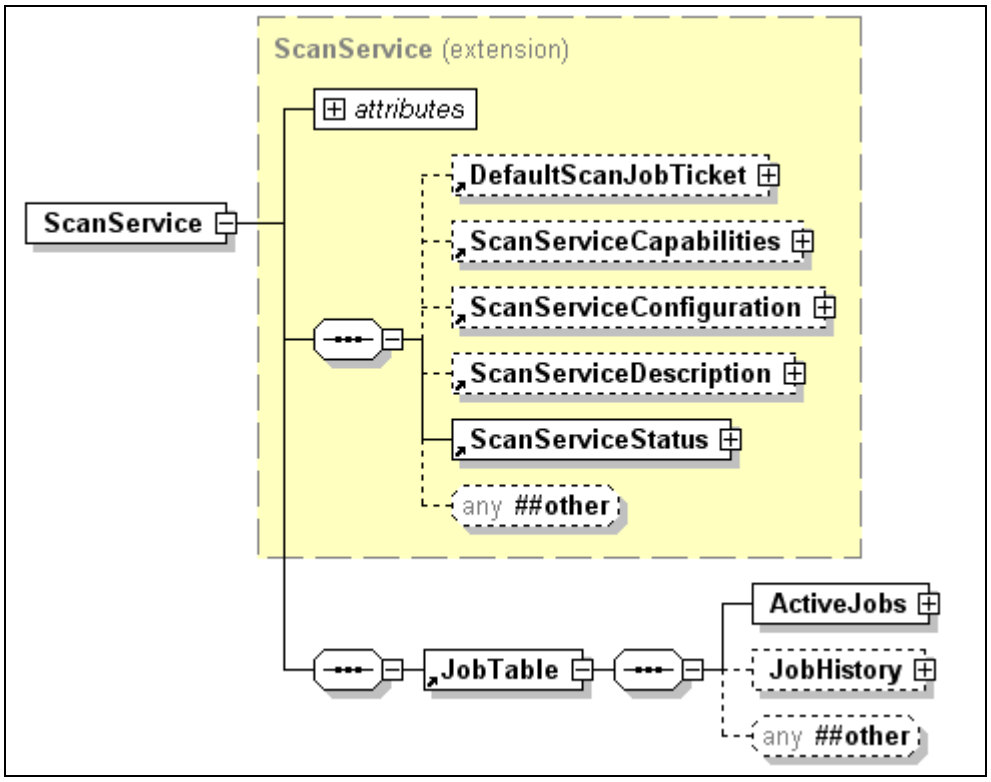


Figure 2 High Level Scan Service Schema

The PWG semantic model supports zero or more Scan Services. A Scan Service is hosted locally on an MFD or remotely on another computer. The Scan Service model has an Active Job queue, a Job History and a set of Elements that includes Scan Service Status, Scan Service Configuration, Scan Service Description and Default Scan Job Ticket.

The ScanServiceStatus Group Element is an extension of the PWG service status class that includes elements such as ID, state, service counters, state messages and state reasons. State messages are localized state reasons. The only

Scan Service specific status extensions are the Scan Service specific counters. The details of the Elements in the ScanServiceStatus group are specified in §7.1.6.

The ScanServiceDescription Group Element includes descriptive information such as service name and information, and has an extension point for vendor specific information. These Description Elements are settable by Administrators. Similar to Scan Service Status elements, there are localized Description Elements for each supported Description Element. The details of the ScanServiceDescription Elements are specified in §7.1.5.

The ScanServiceCapabilities Group Element represents the allowed values supported by the Scan Service for a ScanJobTicket. The Element includes two sub Elements: the ScanDocumentCapabilities Group Element has all processing Elements for Scan Documents, and the ScanJobCapabilites Group Element includes all supported processing Elements for Scan Jobs. The details of each processing Element are specified in §7.1.2.

The DefaultScanTicket Group Element contains scan description and job and document processing default values. The values contained in the Default ScanJobTicket are the values that that will be used by the Scan Service when processing a ScanJobTicket which does not specify an explicit value. The values for this are populated in an implementation specific manner. The details of the DefaultScanTicket are specified in §7.1.1

The ScanServiceConfiguration provides a Scan Service specific view into the Subunits that are associated with this service instance. Only Subunits that are used by the Scan Service will appear in this element. The details of each subunit are detailed in §7.1.3.

A Scan Service contains zero or more Scan Jobs. Each job has zero or more Scan Documents which reference a Destination where the Digital Document(s) are stored as files. The Scan Service organizes its Scan Jobs in a minimum of 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 Scan Jobs that have completed processing.

6.1 Relationships

The diagram below provides a pictorial view of the relationships between the various objects and data items. As mentioned above a Scan Service can contain multiple Scan Jobs. In practice the Jobs appear in one of two queues. One is for pending and active jobs and the other for completed jobs. Each Scan Job can contain multiple Scan Documents. During Job creation there can be a period of time during which a Scan Job does not yet have a Scan Document. It is also possible for a Scan Job to contain multiple Scan Documents. Scan Documents are associated with Digital Documents that contain the data from the scanned Hardcopy Documents. (See §6.2 for a description of the Job/Document/Digital Document cardinality.)

The input to the Scan Service is a set of Hardcopy Documents. A ScanRegion identifies the portion of the Hardcopy Document to be scanned and converted into an Image. It is possible to identify multiple ScanRegions on a media sheet side. Each ScanRegion has a one to one mapping to a scanned Image. The Scan Service uses the Scanner Subunit to scan the ScanRegion and converts the acquired data into an Image. The representation of the Image at this point is implementation specific. The Scan Service accumulates the set of Images that represent the input Hardcopy Document. In an implementation specific manner the set of Images are encoded into the specified Document Format. Document Formats include Image specific formats such as jpeg. The Digital Document which is in the specified Document Format is stored by the Scan Service to the specified Destination. The next section covers the cardinality of the Digital Document.

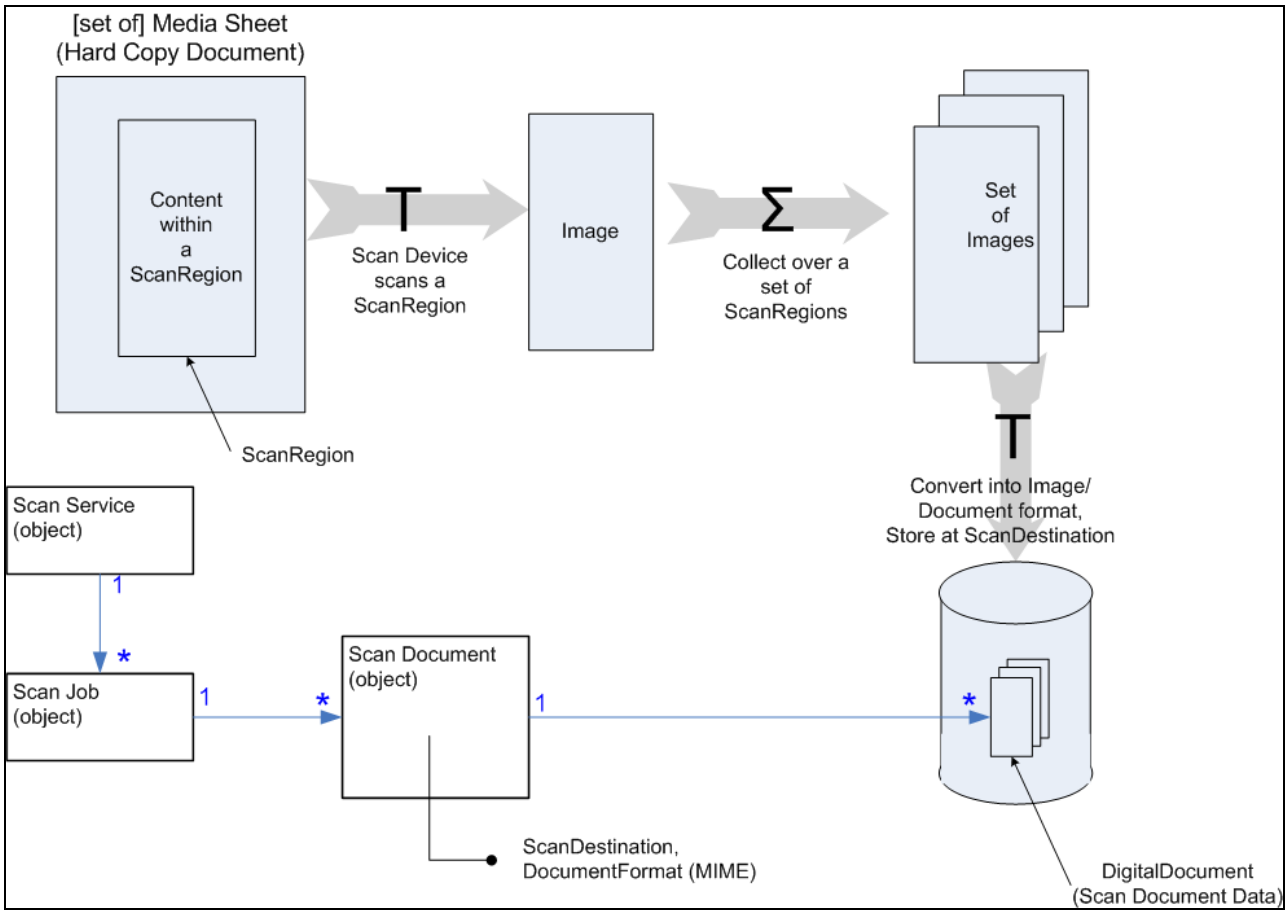


Figure 3 Object Relationships

6.2 Job/Document Object and Digital Document Cardinality

The relationship between input Hardcopy Documents, the Scan Job, Scan Document and Digital Document is described below. The Scan Service semantic model may allow the End User to specify any of the following types of Job, Documents and files (i.e. Digital Document) as the final output of scanning Hardcopy Document(s):

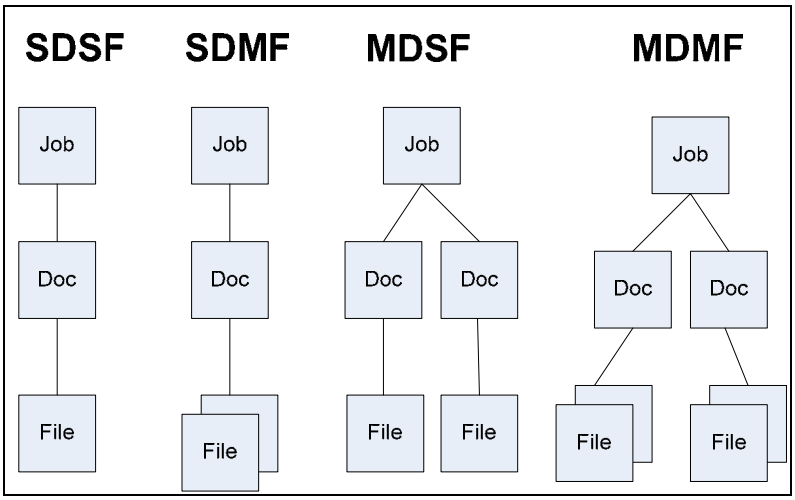


Figure 4 Document Cardinality

- **SDSF** (Single Document Single File Job): One Document object that contains a reference to one single file. There is one file URI for the single Document location. The Destination is the Document URI; i.e. the single file URI.
- **SDMF** (Single Document Multiple File Job): One Document object that contains references to multiple files. Each Image in the Document data is stored in a separate file. There is one file URI per Image file which constitutes the Document. The Destination is the directory URI of the files for the Document.
- **MDSF** (Multiple Document Single File Job): Job object contains multiple Document objects. Each Document can have a different set of processing parameters. Each file contains a specified number of Images for the Document. Each such set of Images is stored in a separate file. The Document object contains one file URI for the Image file location.
- **MDMF** (Multiple Document Multiple File Job): Job object contains multiple Document objects. Each Document can have different set of processing parameters from another. Each Image in the Document data is stored in a separate file. The Document object contains a file URI for each Image file location. The Destination is the directory URI for the Document.

  Note: A completed Scan Job has 1~N Document objects. A Scan Job has a unique ID within a unique Scan Service. The Documents are numbered 1 to N within the Job. The Destination of a multi-document job is a directory for multi-document files. After the Document file is stored, the Document file URI will be updated in the Document object as the Document location.

  Note: In addition to URI, the MIME type of a stored Document MUST be specified by the End User to clearly indicate the content type to be used for transporting the stored Document.

The Scan Service MUST support the type SDSF Job in which the Document data for the only Document in the job is stored in a single file. The capability of a Scan Service to support the various Document/File cardinalities is declared in the DocumentOutputMode in the ScanJobProcessingCapabilities element. The value of the DocumentOutputMode which is set in the ScanJobTicket allows an End User to request one of the supported values. The detail of scan queues and Scan Document are specified in §9.

6.3 ScanJobTicket Lifecycle

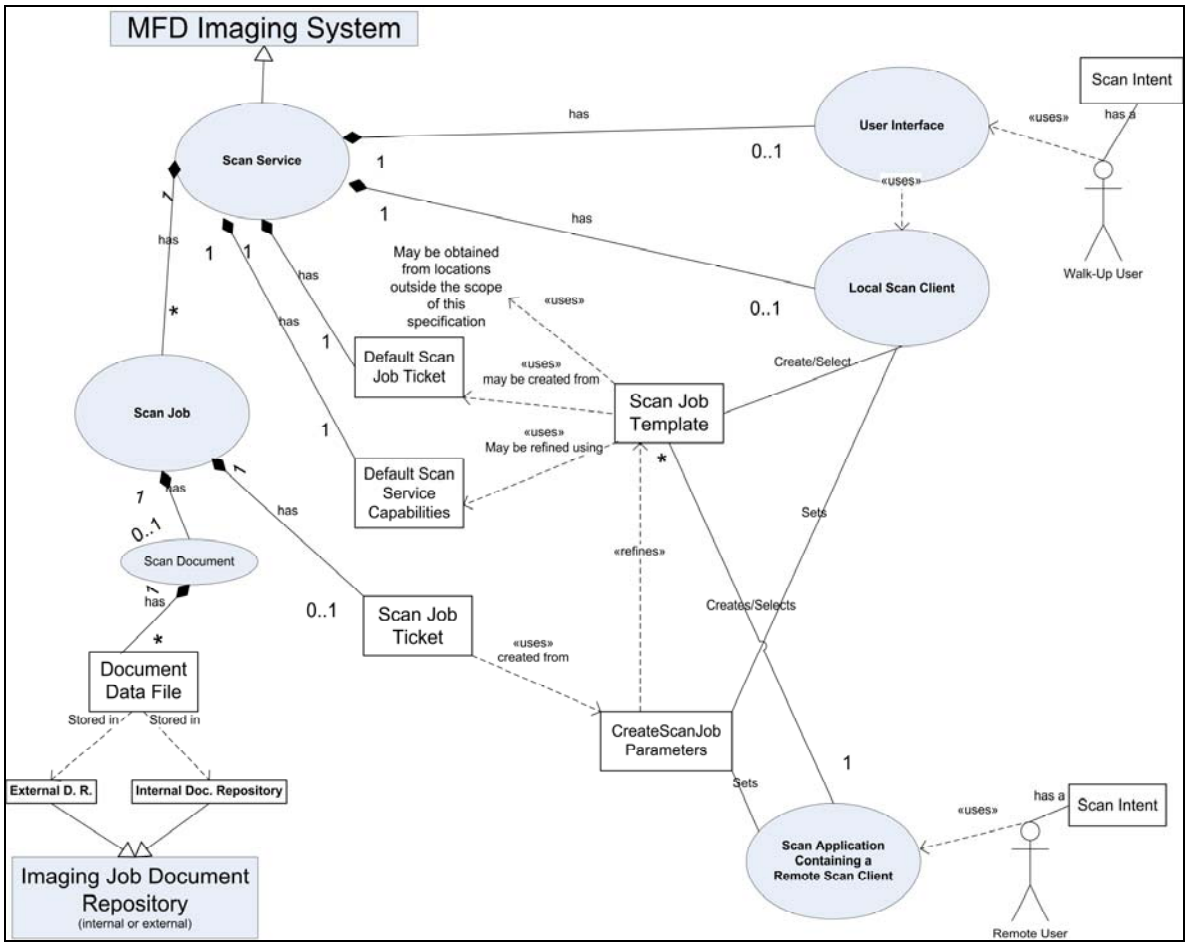


Figure 5 ScanJobTicket Lifecycle Diagram

A ScanJobTicket is a data object that contains job descriptions and job and document processing instructions. The ScanJobTicket life cycle is bound to the lifecycle of its associated Scan Job. The Scan Job is created when a Scan Client initiates a CreateScanJob operation. The CreateScanJob can come from a Remote User interacting with a Scan Application or by a Walkup User interacting with a User Interface and pressing the “start scan” button. In either case the user is able to communicate his Scan Intent through the Scan Service protocol.

Prior to the CreateScanJob submission a ScanJobTemplate can be selected or created. A preconfigured ScanJobTemplate can also be obtained from an external source outside the scope of this specification. Alternatively a ScanJobTemplate can be created by instantiating the associated XML Schema or retrieving the DefaultScanJobTicket from the Scan Service.

The ScanJobTemplate can then be modified to meet the user’s Scan Intent. The Scan Service also has the ScanServiceCapabilities that lists all the allowed values for a Job’s ScanJobTicket for this Scan Service. Once the ScanJobTemplate embodies the Scan Intent to the desired extent, it can be bound to a CreateJob operation parameter to send to the Scan Service.

The Scan Service creates a Scan Job based on the parameters in the CreateScanJob operation. The Scan Job’s ScanJobTicket is created based on the user supplied parameters. The ScanJobTicket remains bound to the Scan Job until the Job is eventually deleted.

A ScanJobTemplate is a ScanJobTicket, unbound to any Scan Job or Scan Service, pre-created by an End User or someone for other End Users, for the convenience or purpose of submitting future Scan Jobs. A ScanJobTemplate can be created in one of the following ways:

(1) A walk-up End User uses Local application and User Interface to create a ScanJobTemplate by configuring a set of preferred values for job and document processing and job description.
(2) An End User uses a remote application (in a computer) to create a ScanJobTemplate by configuring a set of preferred values for job and document processing and job description.
(3) A walk-up End User or Remote user creates a ScanJobTemplate by retrieving the Scan Services DefaultScanTicket and optionally modifying the values.
(4) A walk-up End User or Remote user can retrieve a ScanJobTemplate from an implementation specific location and optionally modifying the values. This method is outside the scope of this specification.
(5) In all the above cases the Scan Services ScanServiceCapabilities can be retrieved to provide the set of allowed values when modifying the ScanJobTemplate

If an End User did not select a Scan Job Template, processing directive defaults to the DefaultScanJobTicket values, which is an Element of the Scan Service. The DefaultScanJobTicket can be administratively modified to control default behaviors for the associated Scan Service.

As a ScanJobTicket is created, it is bound to the End User's Scan Job by the Scan Service. The Scan Service then manages and processes the Scan Job according to the instructions supplied in the ScanJobTicket, including file format of the output Digital Document and the location for storing the Digital Document, be it an internal or an external Document Repository . If a processing instruction in the ScanJobTicket can not be supported, the Scan Service either returns an error or substitutes the instruction with another that best matches the user's intent. The actual values used for processing the job are captured in the ScanJobReceipt of the Job in the Job History queue, which can be queried later.

After a Scan Job is completed, it is moved to the Job History queue. A Scan Job is deleted from the Scan Service when it is deleted from the Job History according to a set of site-specific Job History retention rules.

The ScanJobTicket lifecycle ends as soon as the ScanJobTicket is deleted along with the Scan Job.

6.4 Data Types

The following data types are used in the definitions of the elements within the Scan Service model.

any	Used throughout the model to indicate the location for vendor extension points. Vendors wishing to extend the model may do so at these points.
boolean	Data type consisting of two possible values: 'true' and 'false'
complex	This data type is used to indicate the associated element is a container for multiple elements.
List of complex	This data type is used to indicate the associated element is an unordered set of containers for multiple elements.
dateTime	A string of characters that represent a year, month, day, hour, minute, second and timezone. The data type is derived from the ISO definition [ISO 8601:2000 Second Edition]. The lexical representation is described in [XSD] see <http://www.w3.org/TR/xmlschema-2/#dateTime>
octetString	A sequence of octets encoded in a protocol binding specific safe manner. This syntax type is used for opaque data.
int	A 32 bit signed integer
list of int	An unordered set of 32 bit signed integers
range of int	A complex type consisting of the elements "upperBound" and "lowerBound" which are both 32 bit signed integers
keyword	An ASCII string representing a single enumeration value. The characters that may be used in the keyword are letters, digits, colon(:), hyphen(-) and underscore(_). Vendor extended keywords must be qualified with a prefix of valid characters except the colon(:), followed by

	the colon (:) and then vendors extended keyword. This data type is a subset of XML's NMTOKEN data type [XML] See http://www.w3.org/TR/2000/WD-xml-2e-20000814#NT-Nmtoken
list of keyword	An unordered set of keywords
string	A UTF-8 encoded string of UniCode characters. [rfc3629], [UNICODE]
list of string	An unordered set of strings
URI	A string containing a URI as defined in [rfc3986]
list of URI	An unordered set of URIs
URI Scheme	A URI scheme as defined in [rfc3986] and registered in [rfc4395]
list of URI scheme	An unordered set of URI Schemes

Table 1 Data Types

6.5 Content Coordinate system

Below are descriptions of the coordinate systems and units of measures for the Scanner Subunit and Scan Service. The output of a Scan Service, the Digital Document, is discussed as it relates to the semantic of the layout affecting Document Processing elements. An example would be the rotation of an Image that is stored in a document format that has a "bounding box" in which an Image can be placed. The document format is discussed to provide a consistent set of semantics for the placement of an Image smaller than the "bounding box" or the cropping of an Image larger than the "bounding box".

6.5.1 Scanner Subunit Coordinates

Figure 6 shows the coordinate system that is used for a Scanner Subunit. The content of the media is assumed to be facing the reader. The scan operation begins at the leading edge of the CrossFeed and proceeds down the media. The arrow on the left indicates the movement of the media relative to the scan head. Usually the platen is marked with an arrow that points to the leading edge. The arrow on the platen does not necessarily point to the origin of the coordinate system.

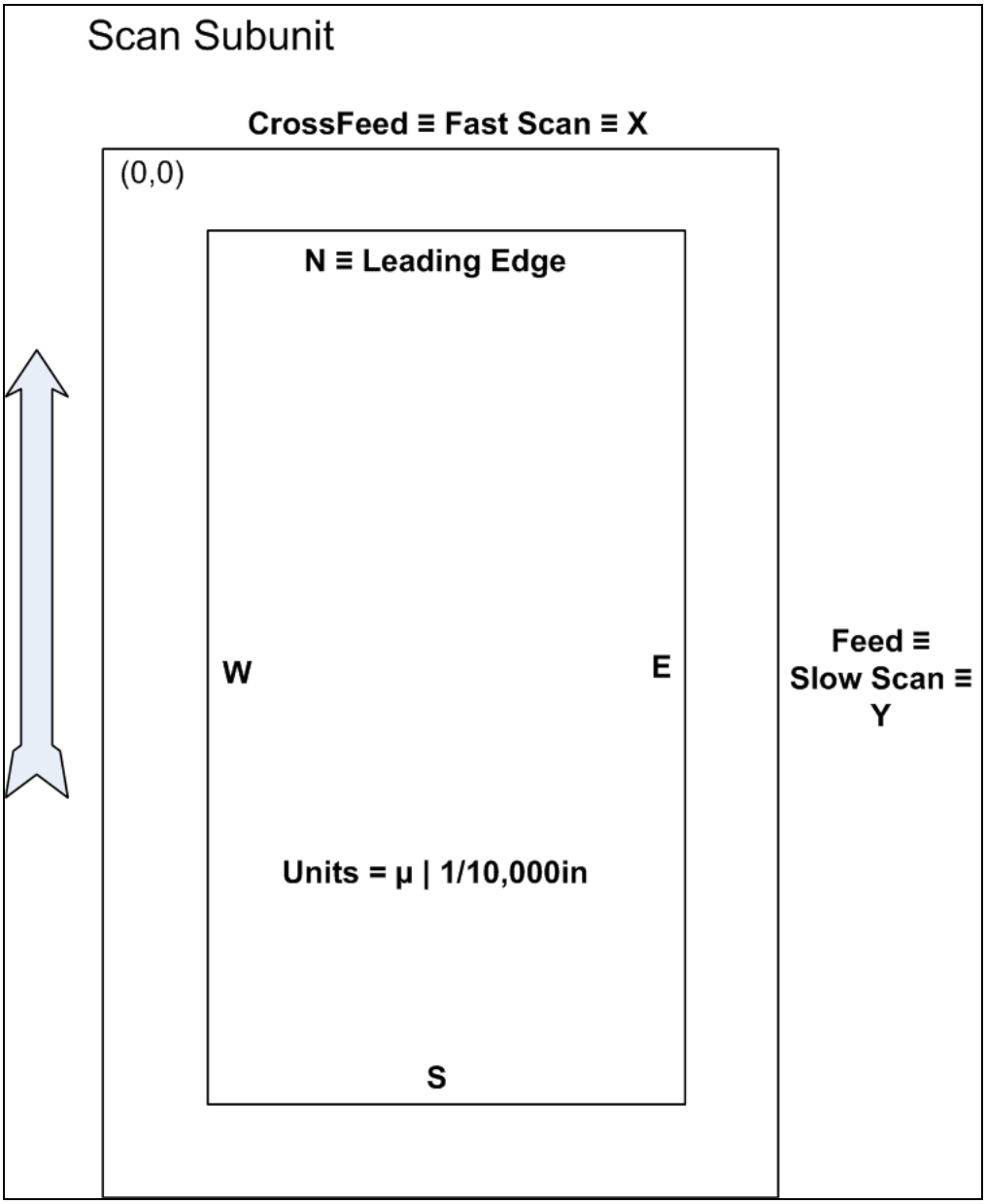


Figure 6 Scanner Subunit Coordinates

6.5.2 Scan Service Coordinates

Figure 7 shows the coordinate system that is used for a Scan Service. The content of the media is assumed to be facing the reader. The scan operation begins at the leading edge of the Fast Scan (i.e. X) direction and proceeds down the media sheet in the Slow Scan (i.e. Y) direction. The coordinate system for the Scanner Subunit and Scan Services are linked together. In other words, the origin of the Scanner Subunit's coordinate system is the same as the origin of the Scan Service's coordinate system. The ScanRegion is specified relative to the origin of the coordinate system. It is possible to have multiple overlapping ScanRegions. The result of a Scan Service scanning a ScanRegion is an Image.

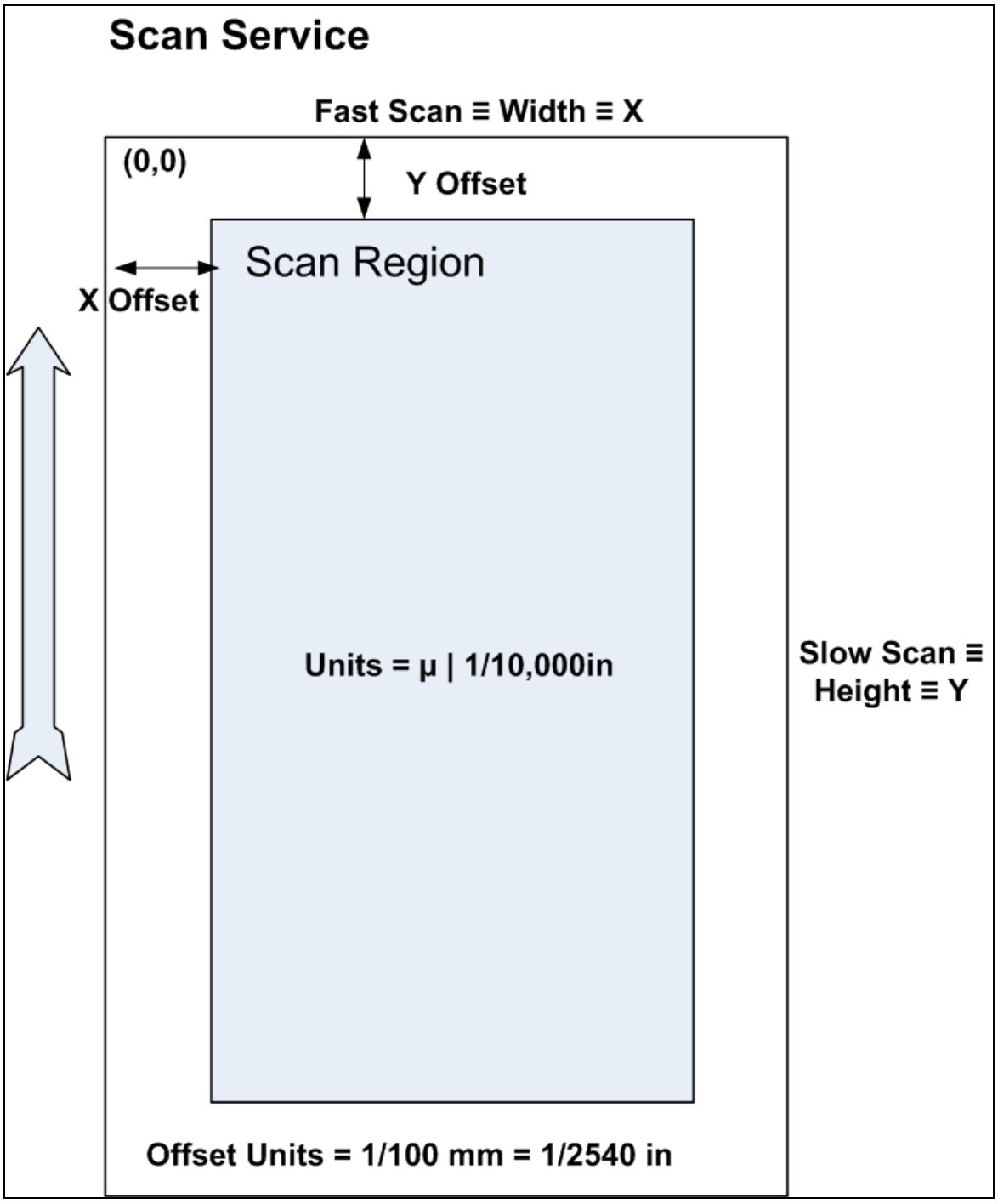


Figure 7 Scan Service Coordinates

6.5.3 Document Format Coordinates

When the output of a Scan Service is stored as an Image file, there is no need to discuss the Document Format coordinates. The Image file contains the entire scanned Image. However when the output of a Scan Service is stored in some Document Formats (e.g. PDF, XPS) the entire Image may not be stored. This is due to the fact that some document formats contains a construct that bounds the area where the Image can be placed.

Figure 8 is the diagram for the Document Format. The image box is where the Image will be placed. The Media Box delineates the area where the Image can appear in the output Document. If the X Offset and Y Offset are absent or set to 0 the origin of the coordinate systems for the Image Box and Media Box are aligned.

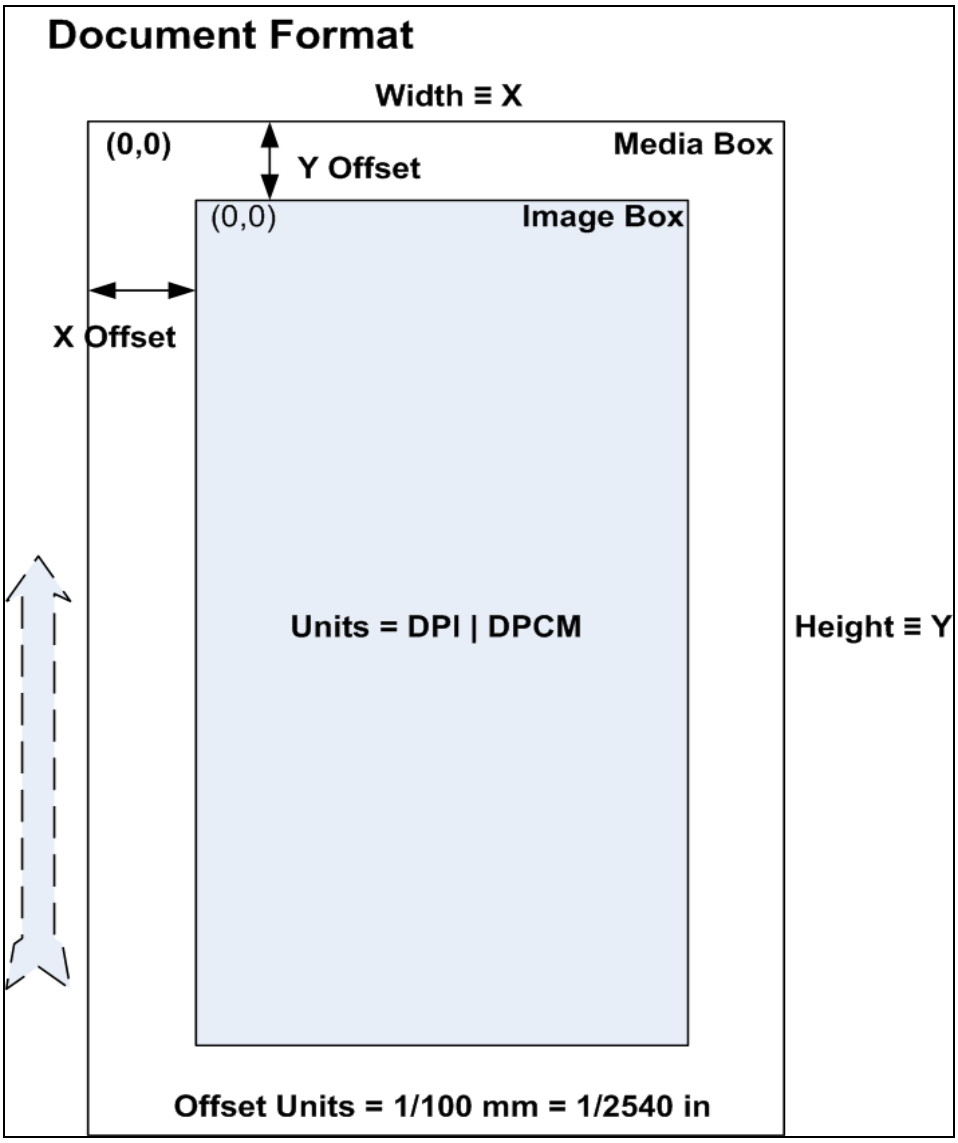


Figure 8 Document Format coordinates

6.5.4 Coordinate Nomenclature

Throughout this specification there appears to be some inconsistency in the names associated with coordinate related semantics. The Scan Service Model is actually a subset of a complete Multifunction Device Model. The goal in a comprehensive multifunction device model is that once a semantic element is defined someplace within the model, it is reused throughout the model. The semantics regarding the coordinate system are affected by this reuse. We have scanner specific semantics (e.g. Fast Scan Direction), Media related semantics (e.g. width), as well as general image processing semantics (e.g. X).

The terms CrossFeed, XFeed, FastScan, X, and Width are semantically aligned. They refer to the axis which is perpendicular to the movement of the Hard Copy Document or Scanner's light bar. It is also the axis along which the Image data is acquired most quickly. The terms Feed, SlowScan, Y, and Height are semantically aligned. They

refer to the axis which is parallel to the movement of the Hard Copy Document or Scanner's light bar. It is also the axis along which the Image data is acquired most slowly.

7 Scan Service Model definition

Below are the detailed descriptions of the elements that comprise the Scan Service model.

7.1.1 DefaultScanJobTicket

(complex – DefaultScanJobTicket) The DefaultScanTicket provides the values that will be used if the element is omitted in a Scan Job's ScanJobTicket. Note that the processing instructions are not bound to the Scan Job until the Scan Job is actually processed. The values from the DefaultScanJobTicket are not copied to the Job's ScanJobTicket. If the ScanJobReceipt is supported, the combined elements from the user supplied ScanJobTicket and the applied values from the DefaultScanJobTicket are copied to the ScanJobReceipt.

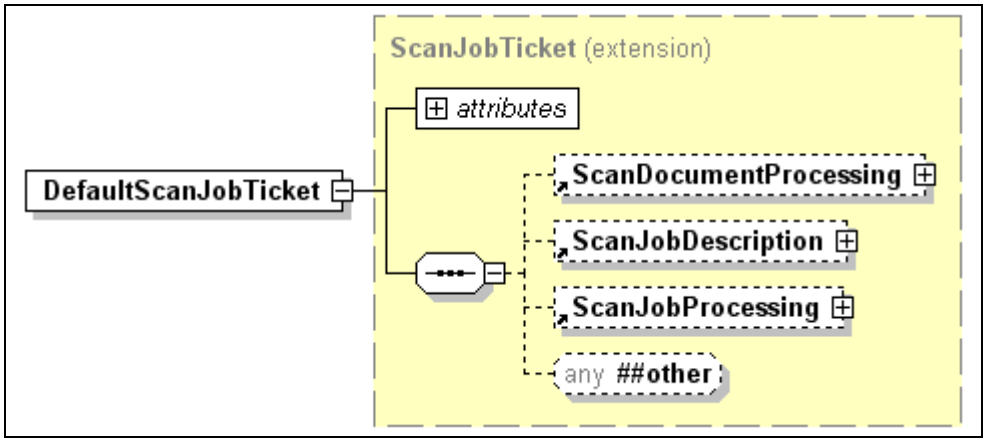


Figure 9 DefaultScanJobTicket

For descriptions of the elements that comprise ScanJobDescription, ScanJobProcessing and ScanDocumentProcessing see §8.1.3 on ScanJobTicket below.

7.1.2 ScanServiceCapabilities

(complex – ScanServiceCapabilities) The ScanServiceCapabilities provides information about the elements that can be used in ScanJobTickets. Although the elements have the same name as their ScanJobTicket counterparts the syntax is often different. For example a ScanJobTicket elements such as InputSource that is a single keyword is a sequence of keywords in ScanServiceCapabilities. The values list the allowed values for the ScanJobTicket element. Some elements that are of the data type integer in a ScanJobTicket are a range of integers in ScanServiceCapabilities. Other elements that are simple strings or predefined ranges in the ScanJobTicket are simply boolean values in Scan Capabilities, indicating supported ScanJobTicket elements.

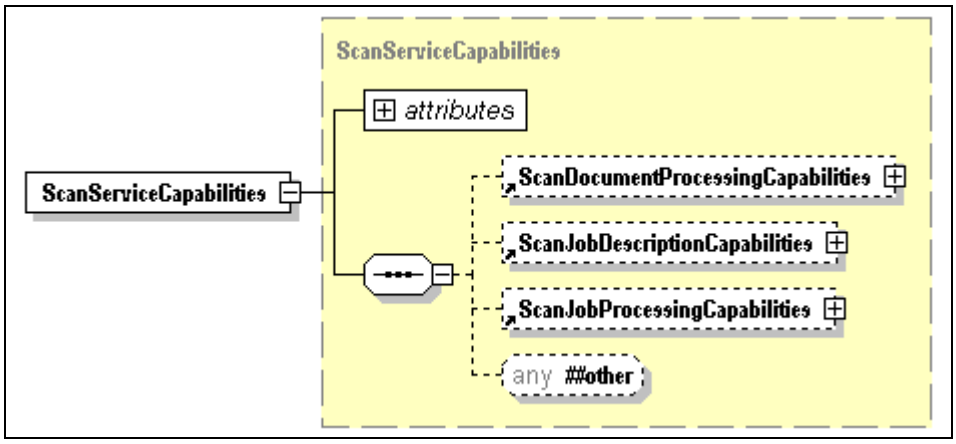


Figure 10 ScanServiceCapabilities

7.1.2.1 Scan Document Processing Capabilities

(complex– ScanDocumentCapabilities) The ScanDocumentCapabilities provides information about the document processing elements that can be used in ScanJobTickets. See §8.1.3.1 for details on semantics of the Scan Document Processing elements.
Note that ScanDocumentCapabilities consists of two sequences. The first represents elements inherited from the Imaging Service super class and includes elements such as NumberUp. The second sequence includes Scan Service specific extensions to the super class such as Exposure and Resolution.

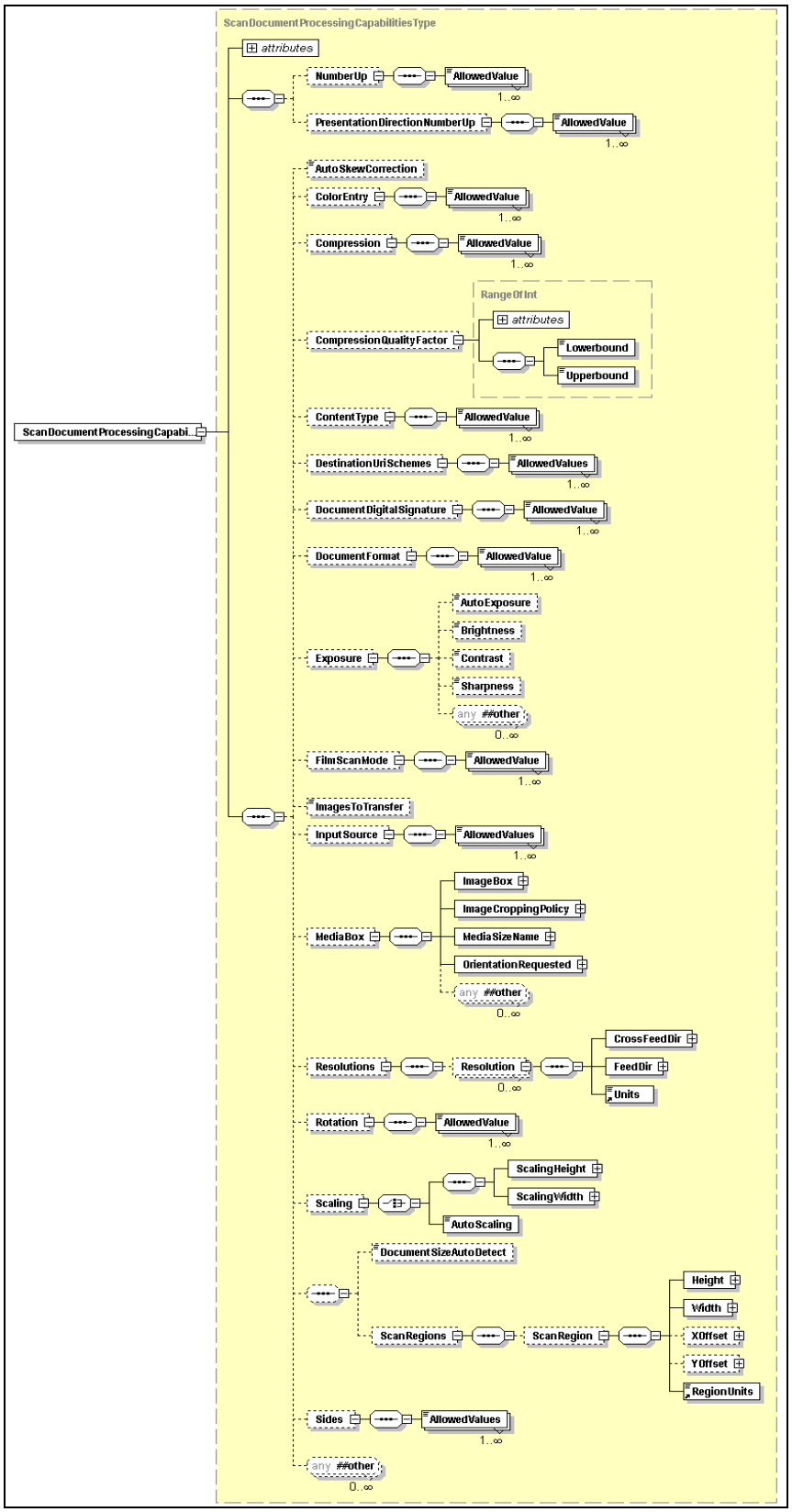


Figure 11 ScanDocumentsProcessingCapabilities

7.1.2.1.1 NumberUp

(list of int) This is a list of numbers that indicates the capability of the Scan Service to assemble multiple Images from the input ScanRegions to be imaged on to the finished output Image. For example, a list of values of '1', '2' and '4' indicates that an output Image can contain the Image from one ScanRegion, two consecutive scan regions or 4 consecutive ScanRegions. If this element is omitted the values are assumed to be limited to 1.

7.1.2.1.2 PresentationDirectionNumberUp

(list of keyword) This element is associated with the "NumberUp" element. It indicates the supported placement order of the Images from the input ScanRegions on to the finished output. If this element is omitted, the values are assumed to be limited to 'TorightTobottom'. *(Keywords: TorightTobottom, TobottomToright, ToleftTobottom, TobottomToleft, TorightTotop, TotopToright, ToleftTotop)*.

7.1.2.1.3 AutoSkewCorrection

(boolean) This element specifies the Scan Service's ability to detect and correct small skew orientation error due to ADF loading or placement on the platen by the user.

7.1.2.1.4 ColorEntry

(list of keyword) This element specifies the color related characteristics for the output of the Scan Service. (See § 8.1.3.1.4) *(Keywords: BlackAndWhite1, Grayscale4, Grayscale8, Grayscale16, RGB24, RGB48, RGBa32, RGBa64, CMYK32, CMYK64)*

7.1.2.1.5 Compression

(list of keyword) This element specifies the compression algorithms supported by the Scan Service. (*Keywords: Compress, Deflate, Gzip, None*)

7.1.2.1.6 CompressionQualityFactor

(Range of Int) This element specifies the range of compression factor supported.

7.1.2.1.7 ContentType

(list of keyword) This element specifies the main characteristics of the original Document supported by the Scan Service. *(Keywords: Auto, Text, Photo, TextAndPhoto, LineArt, Magazine, Halftone)*

7.1.2.1.8 DestinationUriScheme

(list of URI Schemes) This element indicates the supported URI Schemes that can be used to transfer the Digital Document to its Destination. TheSee [rfc4395]

7.1.2.1.9 DocumentDigitalSignature

(list of keywords) This element indicates the types of digital signature that can be used in the creation of the Digital Document. *(Keywords*: *dss, none, pgp, smime, xmldsig)* See [PWG5100.7] §3.2.3 [PWG5100.5] §9.1.11

7.1.2.1.10 DocumentFormat

(list of keyword) This element indicates the output Document formats that can be generated by the Scan Service. The document format values are MIME types. (*Keywords include: application/octet-stream, application/pdf, application/postscript, application/vnd.pwg-xhtml-print+xml, application/vnd.hp-PCL, image/g3fax, image/jpeg, image/tiff, image/tiff-fx*)

7.1.2.1.11 Exposure

(complex) This is a group of elements relating to the perceived quality of a scanned Image

7.1.2.1.11.1 AutoExposure

(boolean) This element indicates whether the Scan Service supports automatic adjustment of Brightness Contrast and/or Sharpness.

7.1.2.1.11.2 Brightness

(boolean) This element indicates whether increase and decrease brightness is supported by the Scan Service.

7.1.2.1.11.3 Contrast

(boolean) This element indicates whether increase and decrease contrast is supported by the Scan Service.

7.1.2.1.11.4 Sharpness

(boolean) This element indicates whether increase and decrease the sharpness of a scanned Images is supported.

7.1.2.1.11.5 Any

This element provides an extension point for vendor differentiation and implementation specific extensions while maintaining interoperability

7.1.2.1.12 FilmScanMode

(list of keyword) This element specifies the supported exposure types for the film to be scanned. *(Keywords: NotApplicable, ColorSlideFilm, ColorNegativeFilm, BlackandWhiteNegativeFilm)*

7.1.2.1.13 ImagesToTransfer

(boolean) This element indicates whether specifying the number of scanned Images in a Document is supported.

7.1.2.1.14 InputSource

(list of keyword) This element indicates the source of the scanned Document supported. *(Keywords: Platen, Feeder, FilmReader)*

7.1.2.1.15 MediaBox

(complex)This element provides the set of allowed values for elements relating to a MediaBox construct. This element is only applicable when the document format used to store the Digital Document has a similar construct that controls the placement and bounding of the scanned Image on the document format specific page Image. See §8.1.3.1.15 for a description of the processing semantics.

7.1.2.1.15.1 ImageBox

(Height, Width, XOffset, YOffset, each is Range of Int, RegionUnits – list of keyword) This element specifies the supported range of area within the input Document boundaries to scan.

7.1.2.1.15.2 ImageCroppingPolicy

(List of keyword) This element indicates the supported ways to handle mismatches in the size of the Image and the intersection of the ImageBox and MediaBox where the Image will be placed.

7.1.2.1.15.3 MediaSizeName

(List of keyword) This element specifies the allowed sizes of the canvas for an Image in the Digital Document expressed as a media size names. See [pwg5101.1] §5

7.1.2.1.15.4 OrientationRequested

(List of keyword) This element indicates the allowed orientation of the MediaBox. See [rfc2911] §4.2.10

7.1.2.1.16 Resolutions

(list of complex) This element specifies the supported sequence of image capture resolutions for.each supported Unit

7.1.2.1.16.1 CrossFeedDir

(list of int) This specifies the allowed values for resolutions, in Units, for the CrossFeed direction. See [rfc2911] §4.2.12

7.1.2.1.16.2 FeedDir

(list of int) This specifies the allowed values for resolutions, in Units, for the Feed direction. See [rfc2911] §4.2.12

7.1.2.1.16.3 Units

(keyword) This specifies the units used to express the resolution *(Keywords: Dpcm, Dpi)* See [rfc2911] §4.2.12

7.1.2.1.17 Rotation

(list of int) This element specifies the supported amount of rotation in degrees clockwise for a scanned Document. Vendors may extend the allowed values (allowed values: 0, 90, 180, 270)

7.1.2.1.18 Scaling

(complex) This element specifies the support for scaling.

7.1.2.1.18.1 ScalingHeight

(range of Int) This element specifies the supported range of the Scaling in the fast scan direction. A value of '100' specifies that no adjustments are made to the scanned Image. Scaling is expressed in 1 percent increments. Values below '100' reduce the magnification and values above increase magnification.

7.1.2.1.18.2 ScalingWidth

(range of Int) This element specifies the supported range of the Scaling in the fast scan direction. A value of '100' specifies that no adjustments are made to the scanned Image. Scaling is expressed in 1 percent increments. Values below '100' reduce the magnification and values above increase magnification.

7.1.2.1.18.3 AutoScaling

(boolean) This element specifies the support for automatically adjusting the Scaling of the image from the Hardcopy Document in an implementation specific manner.

7.1.2.1.19 ScanRegions

(list of complex) This element specifies the support of areas within the Hardcopy Document boundaries to scan. A ScanRegion is specified in each of the supported RegionUnits.

7.1.2.1.19.1.1 Height

(range of int) This element identifies the range of height for the ScanRegion in RegionUnits

7.1.2.1.19.1.2 Width

(range of int) This element identifies the range of width for the ScanRegion in RegionUnits

7.1.2.1.19.1.3 XOffset

(range of int) This element identifies the range of offset, positive or negative, on the X axis from the West Margin of the Scanner Subunit in RegionUnits

7.1.2.1.19.1.4 YOffset

(range of int) This element identifies the range of offset, positive or negative, on the X axis from the West Margin of the Scanner Subunit in RegionUnits

7.1.2.1.19.1.5 RegionUnits

(keyword) This element identifies the units used for specifying the region. *(Keywords: Other, Unknown, TenThousandthsOfInches, Micrometers, Pixels, Percent)*

7.1.2.1.20 DocumentSizesAutodetect

(boolean) This element specifies the support for automatically adjusting the ScanRegion to fit the size of the media in the Hardcopy Document.

7.1.2.1.21 Sides

(list of keyword) This element specifies the supported ways Images can be scanned from the sides of the Hardcopy Document. *(Keywords: OneSided, TwoSidedLongEdge, TwoSidedShortEdge)*

7.1.2.1.22 Any

This element provides an extension point for vendor differentiation and implementation specific extensions while maintaining interoperability.

7.1.2.2 Scan Job Description Capabilities

(complex – ScanJobDescriptionCapabilities) The ScanJobDescriptionCapabilities group element provides information about the Job description elements that can be used in ScanJobTickets. See §8.1.3.2 for details on semantics of the Job Description elements.

Note that ScanJobDescriptionCapabilities consists of two sequences. The first represents elements inherited from the Imaging Service super class and includes elements such as JobName and JobOriginatingUserName. The second sequence includes Scan Service specific extensions to the super class.

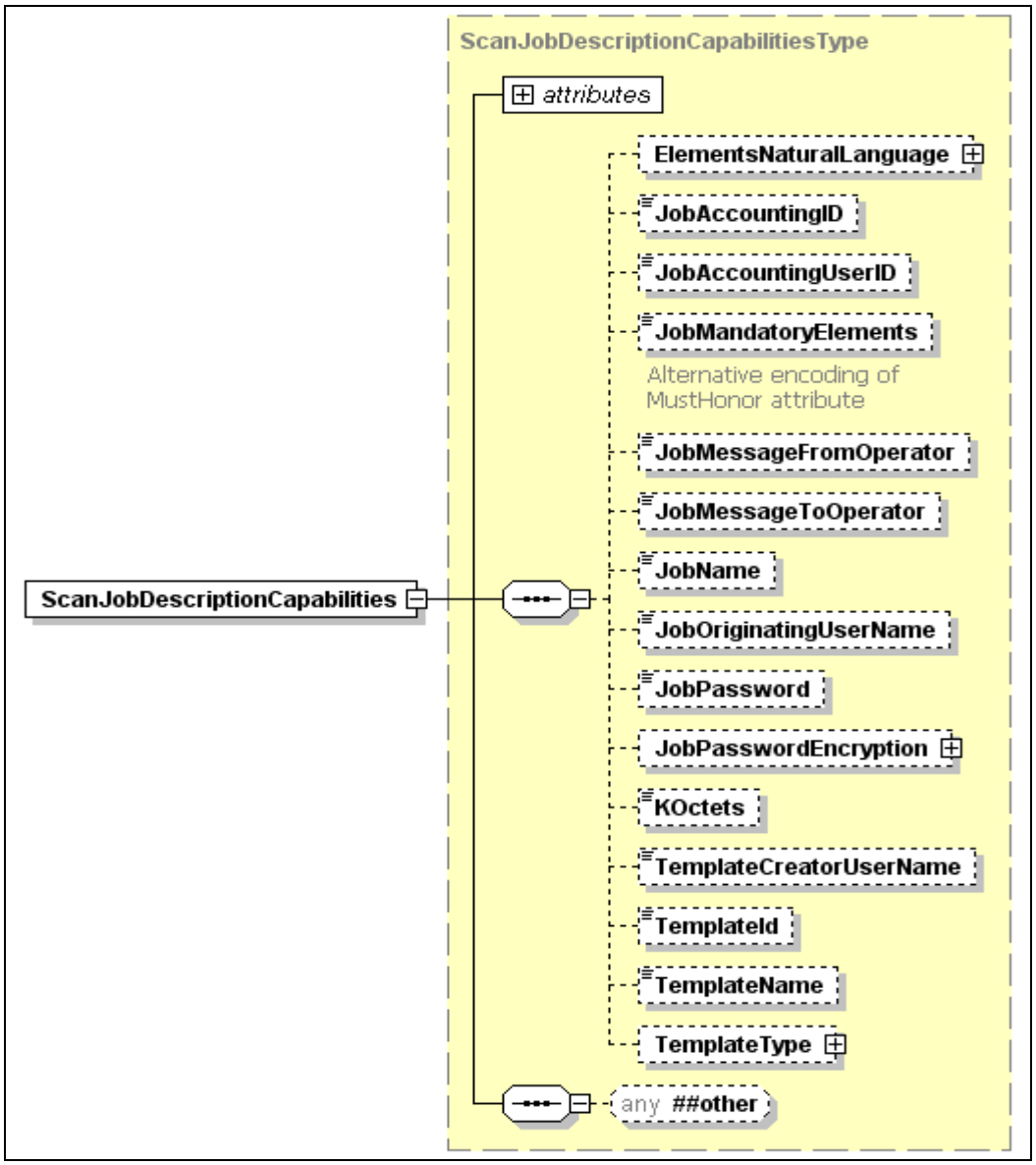


Figure 12 ScanJobDescriptionCapabilities

7.1.2.2.1 ElementsNaturalLanguage

(list of keyword) Indicates the supported natural languages for the elements with a string syntax (example values: en-us, fr, de, ja) See [rfc3066]

7.1.2.2.2 JobAccountingId

(boolean) This element indicates the Scan Service's support for Job Accounts.

7.1.2.2.3 JobAccountingUserId

(boolean) This element indicates the Scan Service's support for a User supplied Accounting ID associated with the Job.

7.1.2.2.4 JobMandatoryElements

(boolean) This element indicates the Scan Service's support JobMandatoryElements which is an alternative method of encoding MustHonor flagged elements in an XML encoding. Therefore this element is not valid in an XML encoding.

7.1.2.2.5 JobMessageFromOperator

(boolean) This element indicates the Scan Service's support of messages from the operator for Jobs.

7.1.2.2.6 JobMessageToOperator

(boolean) This element indicates the Scan Service's support of messages to the operator for Jobs.

7.1.2.2.7 JobName

(boolean) This element indicates the Scan Service's support of a name for the Job.

7.1.2.2.8 JobOriginatingUserName

(boolean) This element indicates the Scan Service's support of the name of the user submitting the Job.

7.1.2.2.9 JobPassword

(boolean) This element indicates the Scan Service's support for holding a Job until the correct password is entered.

7.1.2.2.10 JobPasswordEncryption

(list of keyword) This element contains the list of password encryption and hash methods supported by the Scan Service. (*Keywords: TripleDES, AES, ECC, Md2, Md4, Md5, Sha, None*) See [DES], [ECC], [AES], [MD2], [MD4], [MD5], [SHA]

7.1.2.2.11 KOctets

(boolean) This element indicates the Scan Service's support for storing the size of the Job in integral units of 1024 octets.

7.1.2.2.12 TemplateCreatorUserName

(boolean) This element indicates the Scan Service's support for storing the user who created the template on which the ScanJobTicket was based.

7.1.2.2.13 TemplateId

(boolean) This element indicates the Scan Service's support for storing the Id of the template on which the ScanJobTicket was based.

7.1.2.2.14 TemplateInfo

(boolean) This element indicates the Scan Service's support for storing information about the template on which the ScanJobTicket was based.

7.1.2.2.15 TemplateName

(boolean) This element indicates the Scan Service's support for storing the user friendly name of the template on which the ScanJobTicket was based.

7.1.2.2.16 TemplateType

(list of keyword) This element indicates the Scan Service's support for storing the type of the template on which the ScanJobTicket was based. The Scan Service should only support Scan Job and Scan Document Templates. (*Keywords: ScanJob, ScanDocument*)

7.1.2.2.17 Any

This element provides an extension point for vendor differentiation and implementation specific extensions while maintaining interoperability.

7.1.2.3 Scan Job Processing Capabilities

(complex – ScanJobProcessingCapabilities) The ScanJobProcessingCapabilities group element provides information about the Job processing elements that can be used in ScanJobTickets. See §8.1.3.3 for details on semantics of the Job Processing elements.

Note that ScanJobProcessingCapabilities consists of two sequences. The first represents elements inherited from the Imaging Service super class and includes elements such as JobHoldUntil. The second sequence includes Scan Service specific extensions to the super class such as BatchMode.

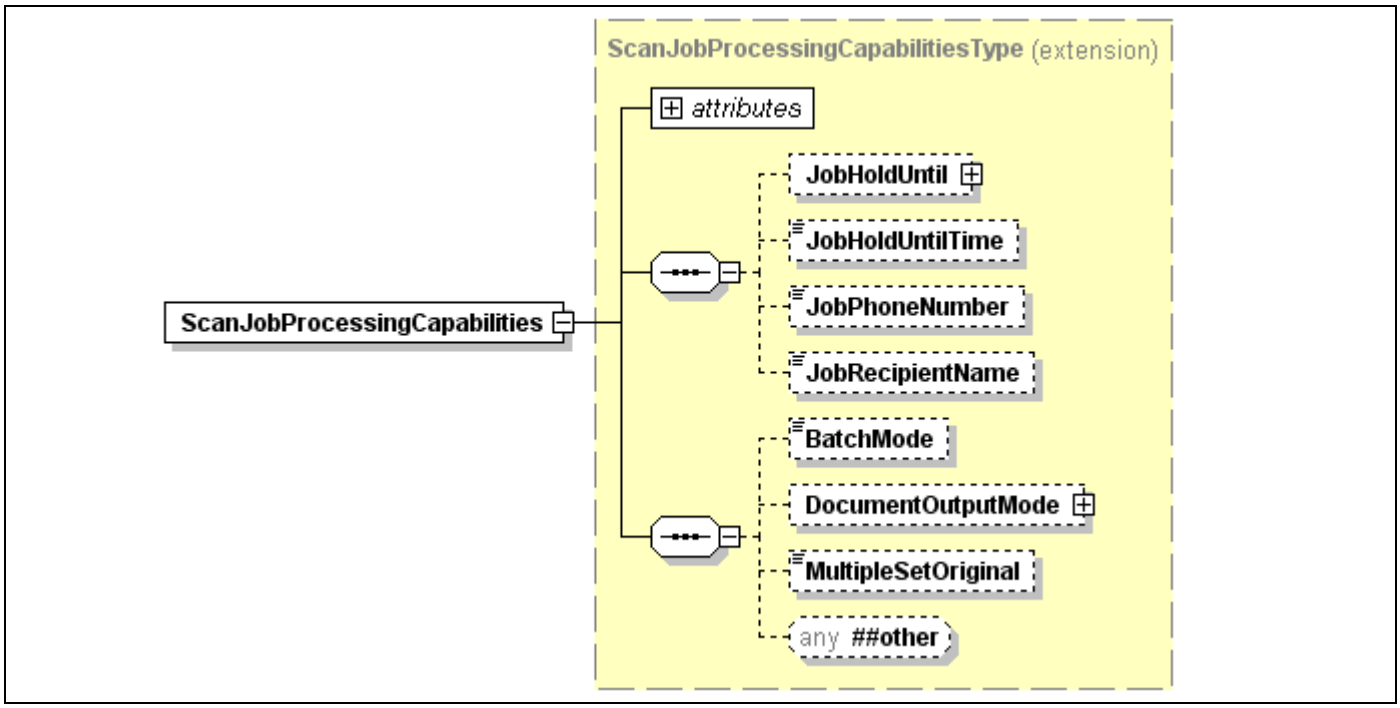


Figure 13 ScanJobProcessingCapabilities

7.1.2.3.1 JobHoldUntil

(list of keyword) This element indicates the named events for releasing the held job. (*Keywords: DayTime, Evening, Indefinite, Night, NoHold, SecondShift, ThirdShift, Weekend*)

7.1.2.3.2 JobHoldUntilTime

(boolean) This element indicates if the Scan Service allows a ScanJobTicket to specify a Date and Time when a job is put on hold will be released.

7.1.2.3.3 JobPhoneNumber

(boolean) This element indicates if storing the contact phone number for the job is supported.

7.1.2.3.4 JobRecipientName

(boolean) This element indicates if storing the name of the recipient of a job is supported.

7.1.2.3.5 BatchMode

(boolean) This element indicates whether Batch Scan processing mode is supported by the Scan Service.

7.1.2.3.6 DocumentOutputMode

(list of keyword) This element indicates the number of Documents and Image storage files combinations supported by the Scan Service. (*Keywords: SingleDocumentSingleFile, SingleDocumentMultipleFile, MultipleDocumentSingleFile, MultipleDocumentMultipleFile*)

7.1.2.3.7 MultipleSetOriginals

(boolean) This element indicates if the Scan Service supports scanning multiple sets of Hardcopy Documents for the Job.

7.1.2.3.8 Any

This element provides an extension point for vendor differentiation and implementation specific extensions while maintaining interoperability.

7.1.3 ScanServiceConfiguration

(complex – ScanServiceConfiguration) This group element identifies the actual instances of the subunits the service uses. Note that not all subunits are used by the Scan Services. Although the diagram shows all the defined Subunits only the applicable subunits are included below.

These elements are based on the Printer MIB [RFC3805]. See the RFC for details on these elements.

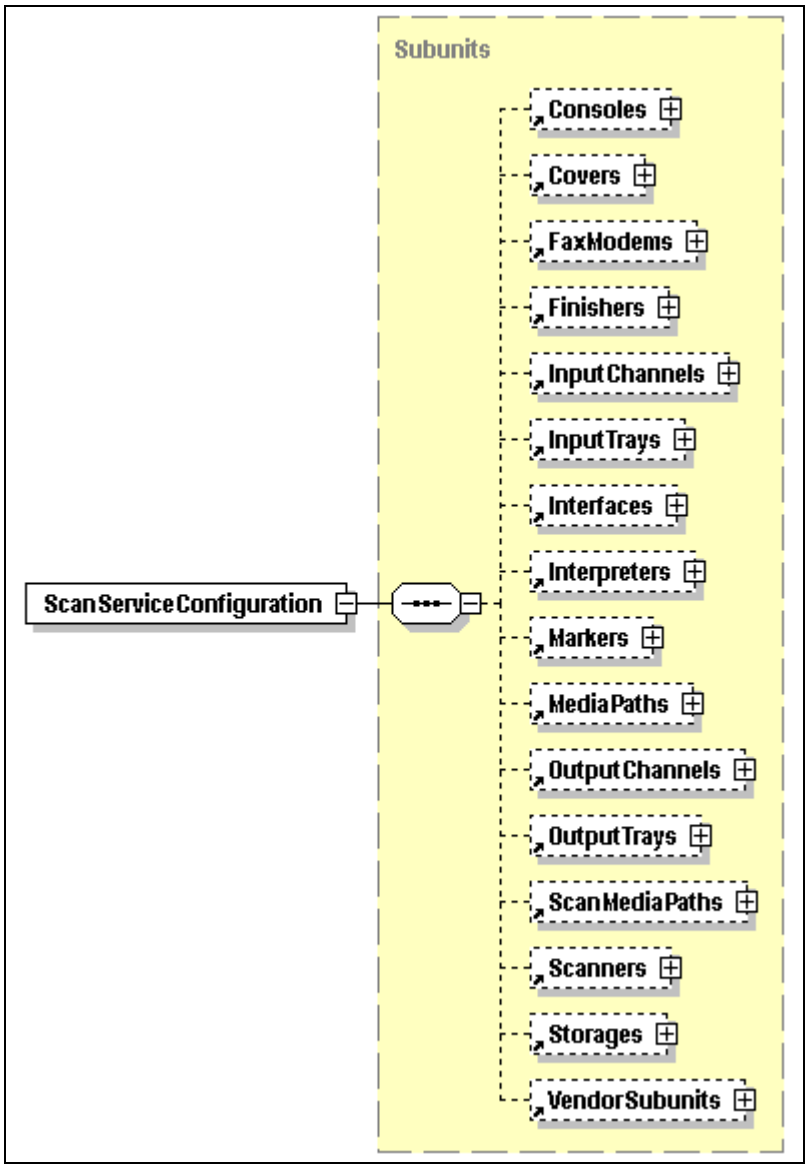


Figure 14 ScanServiceConfiguration

7.1.3.1 Console

(complex – Console) This element represents the User Interface Console(s). See [RFC3805] prtConsoleGroup for details.

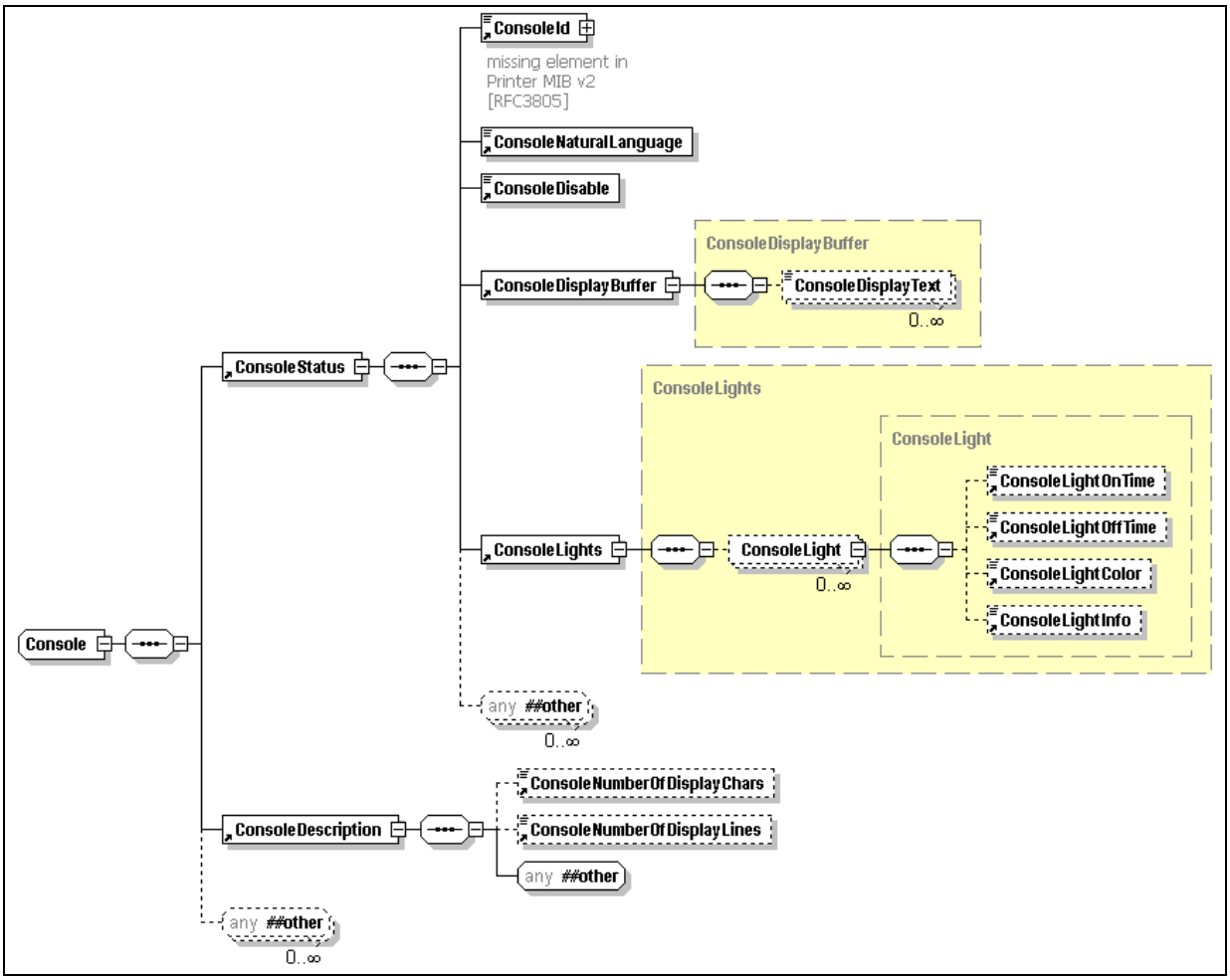


Figure 15 Console

7.1.3.1.1 ConsoleID

(int) This element uniquely identifies an instance of a console.

7.1.3.1.2 ConsoleNaturalLanguage

(string) The language, country, and character set to be used for the console. See [RFC3805] prtConsoleLocalization and [RFC3066] for additional details.

7.1.3.1.3 ConsoleDisable

(boolean) This element indicates if input is accepted from the operator console. See [RFC3805] prtConsoleDisable for additional details.

7.1.3.1.4 ConsoleDisplayBuffer

(complex - ConsoleDisplayBuffer) Physical display buffer for printer console display or operator panel. See [RFC3805] prtConsoleDisplayBufferTable for additional details.

7.1.3.1.4.1 ConsoleDisplayText

(list of string) Buffer associated with each physical line on the display or operator panel. See [RFC3805] prtConsoleDisplayBufferEntry for additional details.

7.1.3.1.5 ConsoleLight

(list of complex – ConsoleLight) Provides a description and state information for each light present on the printer console. See [RFC3805] prtConsoleLightTable and prtConsoleLightEntry for additional details.

7.1.3.1.6 ConsoleLightOnTime

(int) This object, in conjunction with ConsoleLightOffTime, defines the current status of the light. If both ConsoleLightOnTime and ConsoleLightOffTime are non-zero, the lamp is blinking and the values presented define the on time and off time, respectively, in milliseconds. If ConsoleLightOnTime is zero and ConsoleLightOffTime is non-zero, the lamp is off. If ConsoleLightOffTime is zero and ConsoleLightOnTime is non-zero, the lamp is on. If both values are zero the lamp is off. See [RFC3805] prtConsoleOnTime for additional details.

7.1.3.1.7 ConsoleLightOffTime

(int) This object, in conjunction with ConsoleLightOnTime, defines the current status of the light. If both ConsoleLightOnTime and ConsoleLightOffTime are non-zero, the lamp is blinking and the values presented define the on time and off time, respectively, in milliseconds. If ConsoleLightOnTime is zero and ConsoleLightOffTime is non-zero, the lamp is off. If ConsoleLightOffTime is zero and ConsoleLightOnTime is non-zero, the lamp is on. If both values are zero the lamp is off. See [RFC3805] prtConsoleOffTime for additional details.

7.1.3.1.8 ConsoleLightColor

(keyword) The color of this light. (*Keywords: Other, Unknown, White, Red, Green, Blue, Cyan, Magenta, Yelow, Orange*) See [RFC3805] prtConsoleColor for additional details.

7.1.3.1.9 ConsoleLightInfo

(string) The localized vendor description or label of this light. See [RFC3805] prtConsoleDescription for additional details.

7.1.3.1.10 ConsoleNumberOfDisplayChars

(int) "The number of characters per line displayed on the physical display. See [RFC3805] prtConsoleNumberOfDisplayChars for additional details.

7.1.3.1.11 ConsoleNumberOfDisplayLines

(int) The number of lines on the printer's physical display. See [RFC3805] prtConsoleNumberOfDisplayLines for additional details.

7.1.4 Cover

(complex – Cover) This element represents a device's covers, doors, housings or interlocks. See [RFC3805] prtCoverTable for details.

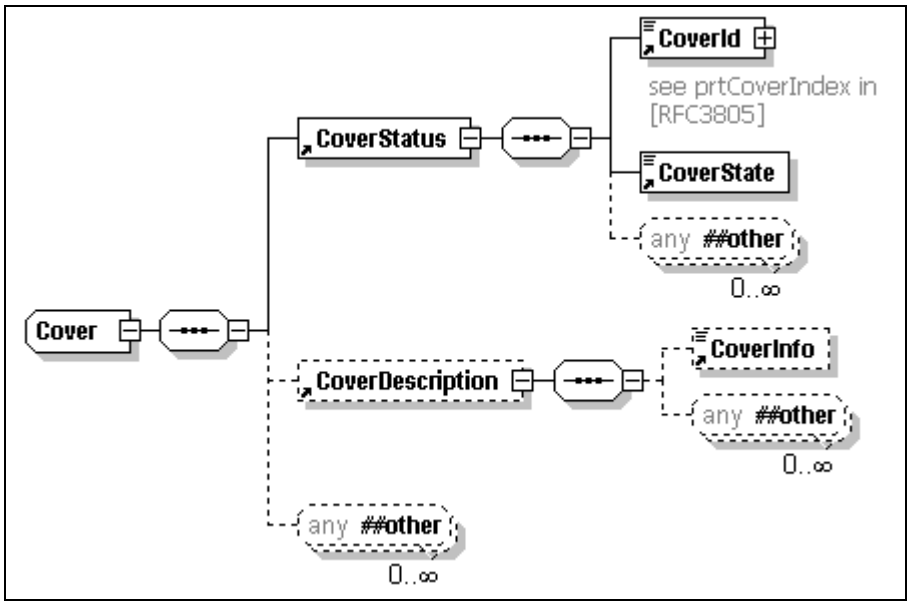


Figure 16 Covers

7.1.4.1.1 CoverID

(int) This element uniquely identifies an instance of a console. See [RFC3805] prtCoverIndex for additional details.

7.1.4.1.2 CoverState

(keyword) This status of this console subunit. (*Keywords: other, coverOpen, coverClosed, interlockOpen, interlockClosed*) See [RFC3805] §6 prtCoverStatus for additional details.

7.1.4.1.3 CoverInfo

(string) The manufacturer provided localized name for the cover subunit. See [RFC3805] prtCoverDescription for additional details.

7.1.4.2 FaxModem

(complex – FaxModem) Not applicable to a Scan Service. See [RFC1696] mdmMib for additional details.

7.1.4.3 Finisher

(complex – Finisher) Not applicable to a Scan Service. See [RFC3806] finDevice for details.

7.1.4.4 InputChannel

(complex – InputChannel) This element represents the source of Scan Job control information (i.e. channel over which Scan Service operations are received). Note that InputChannelDefaultPageDescriptionLanguage does not apply to Scan Service. See [RFC3805] prtChannelTable for additional details.

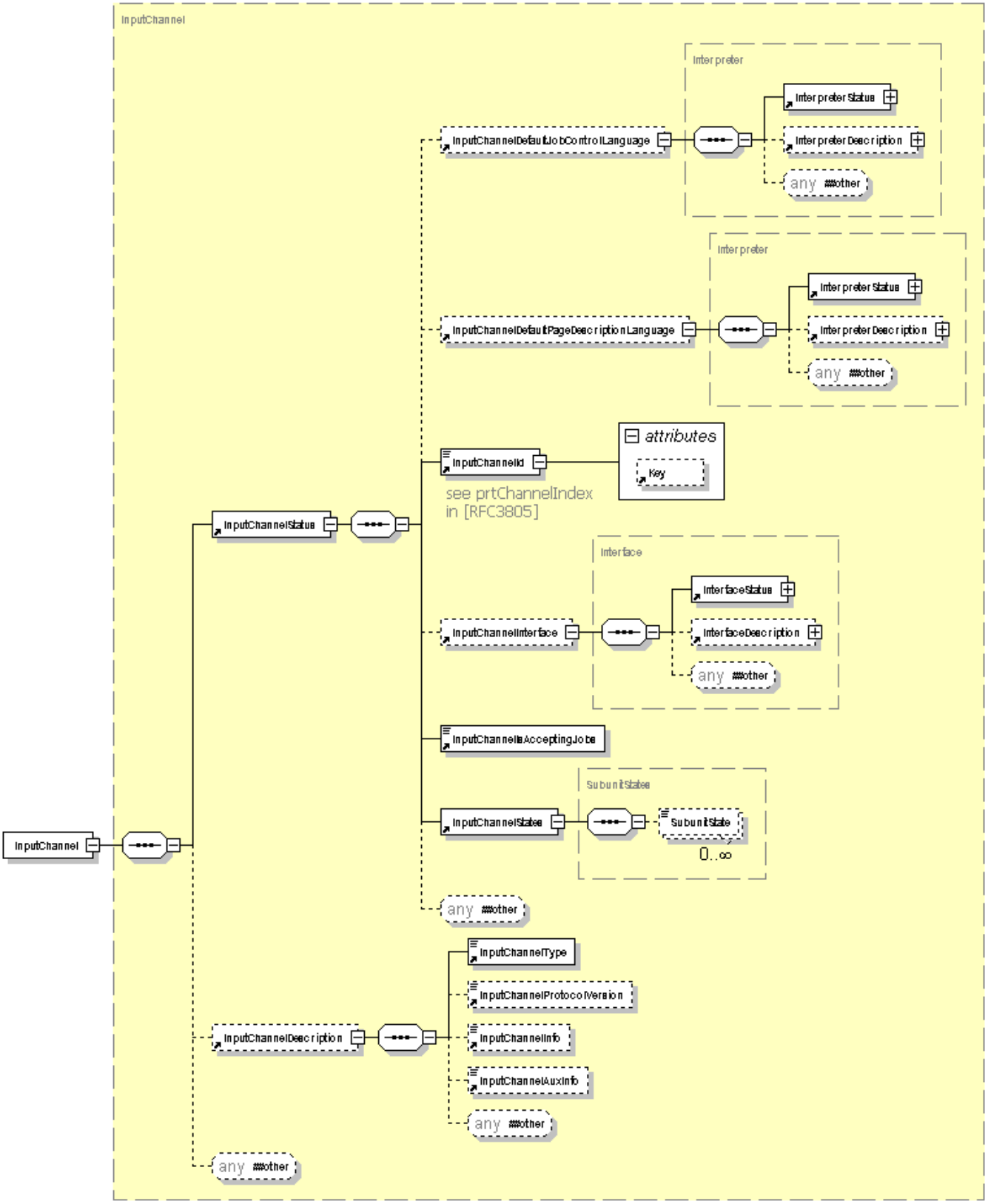


Figure 17 InputChannel

7.1.4.5 InputTrays

(complex – InputTrays) Not applicable to a Scan Service. See [RFC3805] prtInputTable for additional details.

7.1.4.6 Interfaces

(complex – Interfaces) This element represents the communication ports and associated protocol stack into and out of the device. The details of the elements contained within this element group are defined in [RFC1213] and [RFC2863] ifTable.

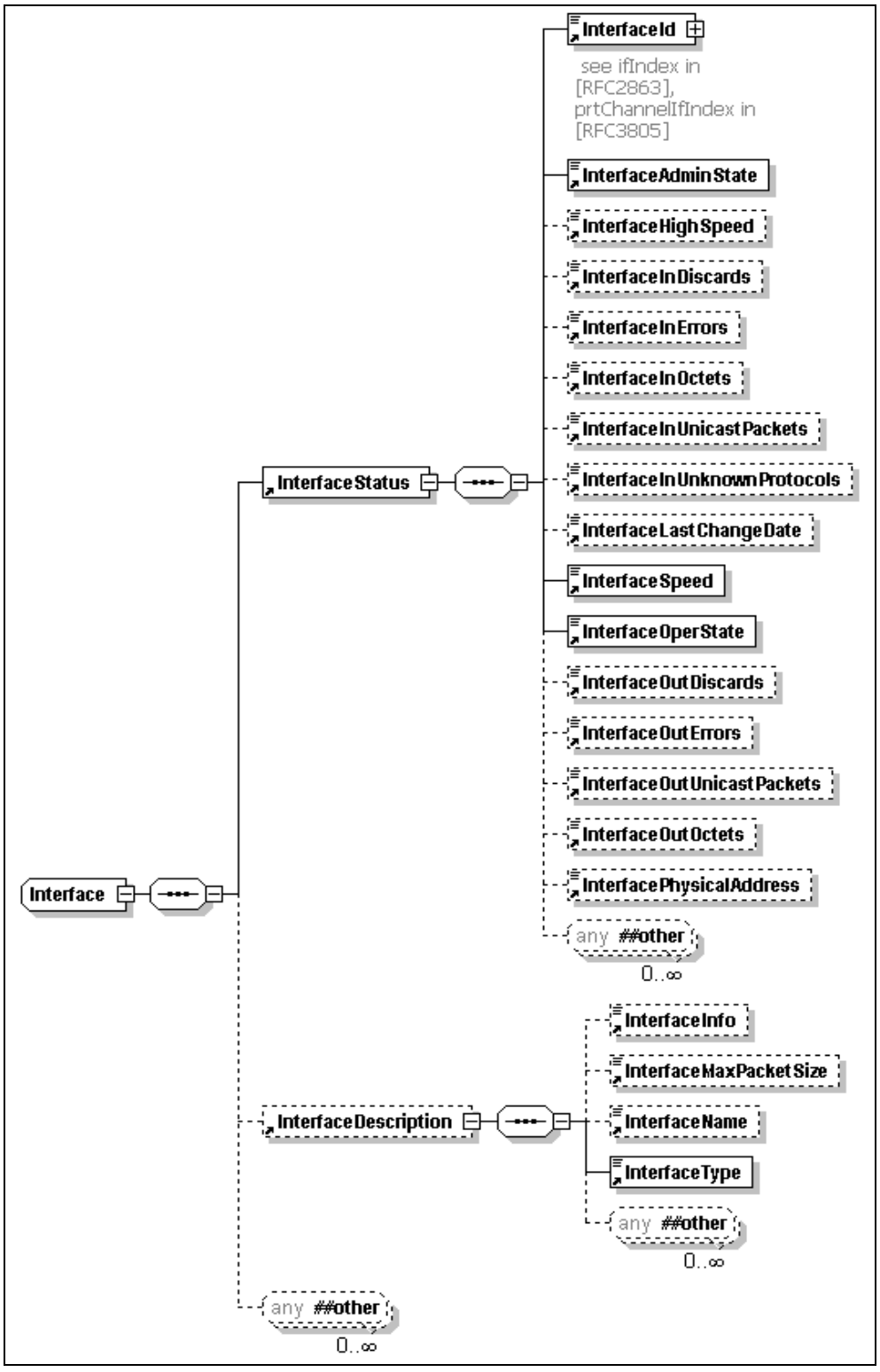


Figure 18 Interface

7.1.4.6.1 InterfaceID

(int) This element uniquely identifies an instance of an interface. See [RFC2863] ifIndex for additional details.

7.1.4.6.2 InterfaceAdminState

(keyword) The desired state of the interface. (*Keywords: Up, Down, Testing*) See [RFC2863] ifAdminStatus for additional details.

7.1.4.6.3 InterfaceHighSpeed

(int) An estimate of the interface's current bandwidth in units of 1,000,000 bits per second. See [RFC2863] ifHighSpeed for additional details.

7.1.4.6.4 InterfaceInDiscards

(int) The number of inbound packets which were chosen to be discarded even though no errors had been detected to prevent their being deliverable to a higher-layer protocol. See [RFC2863] ifInDiscards for additional details.

7.1.4.6.5 InterfaceInErrors

(int) For packet-oriented interfaces, the number of inbound packets that contained errors preventing them from being deliverable to a higher-layer protocol. For characteroriented or fixed-length interfaces, the number of inbound transmission units that contained errors preventing them from being deliverable to a higher-layer protocol. See [RFC2863] ifInErrors for additional details.

7.1.4.6.6 InterfaceInOctets

(int) The total number of octets received on the interface, including framing characters. See [RFC2863] ifInOctets for additional details.

7.1.4.6.7 InterfaceInUnicastPackets

(int) "The number of packets, delivered by this sub-layer to a higher (sub-)layer, which were not addressed to a multicast or broadcast address at this sub-layer. See [RFC2863] ifInUcastPkts for additional details.

7.1.4.6.8 InterfaceInUnknownProtocols

(int) For packet-oriented interfaces, the number of packets received via the interface which were discarded because of an unknown or unsupported protocol. For character-oriented or fixed-length interfaces that support protocol multiplexing the number of transmission units received via the interface which were discarded because of an unknown or unsupported protocol. See [RFC2863] ifInUnknownProtos for additional details.

7.1.4.6.9 InterfaceLastChangeDate

(dateTime) The date/time the interface entered its current operational state. See [RFC2863] ifLastChange for additional details. NOTE: The syntax of this element (DateTime) differs from its MIB counterpart (int i.e. seconds).

7.1.4.6.10 InterfaceSpeed

(int) An estimate of the interface's current bandwidth in bits per second. If the bandwidth of the interface is greater than the maximum value reportable by this object then this object SHOULD report its maximum value (4,294,967,295) and InterfaceHighSpeed MUST be used to report the interface's speed. See [RFC2863] ifSpeed for additional details.

7.1.4.6.11 InterfaceOperState

(keyword) The current operational state of the interface. (*Keywords: Up, Down, Testing, Unknown, Dormant, NotPresent, LowerLayerDown*) See [RFC2863] for additional details on ifIndex.

7.1.4.6.12 InterfaceOutDiscards

(int) The number of outbound packets which were chosen to be discarded even though no errors had been detected to prevent their being transmitted. See [RFC2863] ifOutDiscards for additional details.

7.1.4.6.13 InterfaceOutErrors

(int) For packet-oriented interfaces, the number of outbound packets that could not be transmitted because of errors. For character-oriented or fixed-length interfaces, the number of outbound transmission units that could not be transmitted because of errors. See [RFC2863] ifOutErrors for additional details.

7.1.4.6.14 InterfaceOutUnicastPackets

(int) The total number of packets that higher-level protocols requested be transmitted, and which were not addressed to a multicast or broadcast address at this sub-layer, including those that were discarded or not sent. See [RFC2863] ifOutUcastPkts for additional details.

7.1.4.6.15 InterfaceOutOctets

(int) The total number of octets transmitted out of the interface, including framing characters. See [RFC2863] ifOutOctets for additional details.

7.1.4.6.16 InterfacePhysicalAddress

(string) The interface's address at its protocol sub-layer. See [RFC2863] ifPhysAddress for additional details.

7.1.4.6.17 InterfaceInfo

(string) A textual string containing information about the interface. See [RFC2863] ifDescr for additional details.

7.1.4.6.18 InterfaceMaxPacketSize

(int) The size of the largest packet which can be sent/received on the interface, specified in octets. See [RFC2863] ifMtu for additional details.

7.1.4.6.19 InterfaceName

(string) The textual name of the interface. The value of this object should be the name of the interface as assigned by the local device and should be suitable for use in commands entered at the device's 'console'. See [RFC2863] ifName for additional details.

7.1.4.6.20 InterfaceType

(int) The type of interface. Values are assigned by the Internet Assigned Numbers Authority (IANA), through updating the syntax of the IANAifType textual convention. See [RFC2863] ifType and [IANA_NUMBERS] for additional details.

7.1.4.7 Interpreters

(complex – Interpreters) Applicable to Scan Service for two purposes. One is to indicate a control language associated with an output channel. The other is to describe the formatting subunit for the output digital document. See [RFC3805] prtInterpreterTable for additional details.

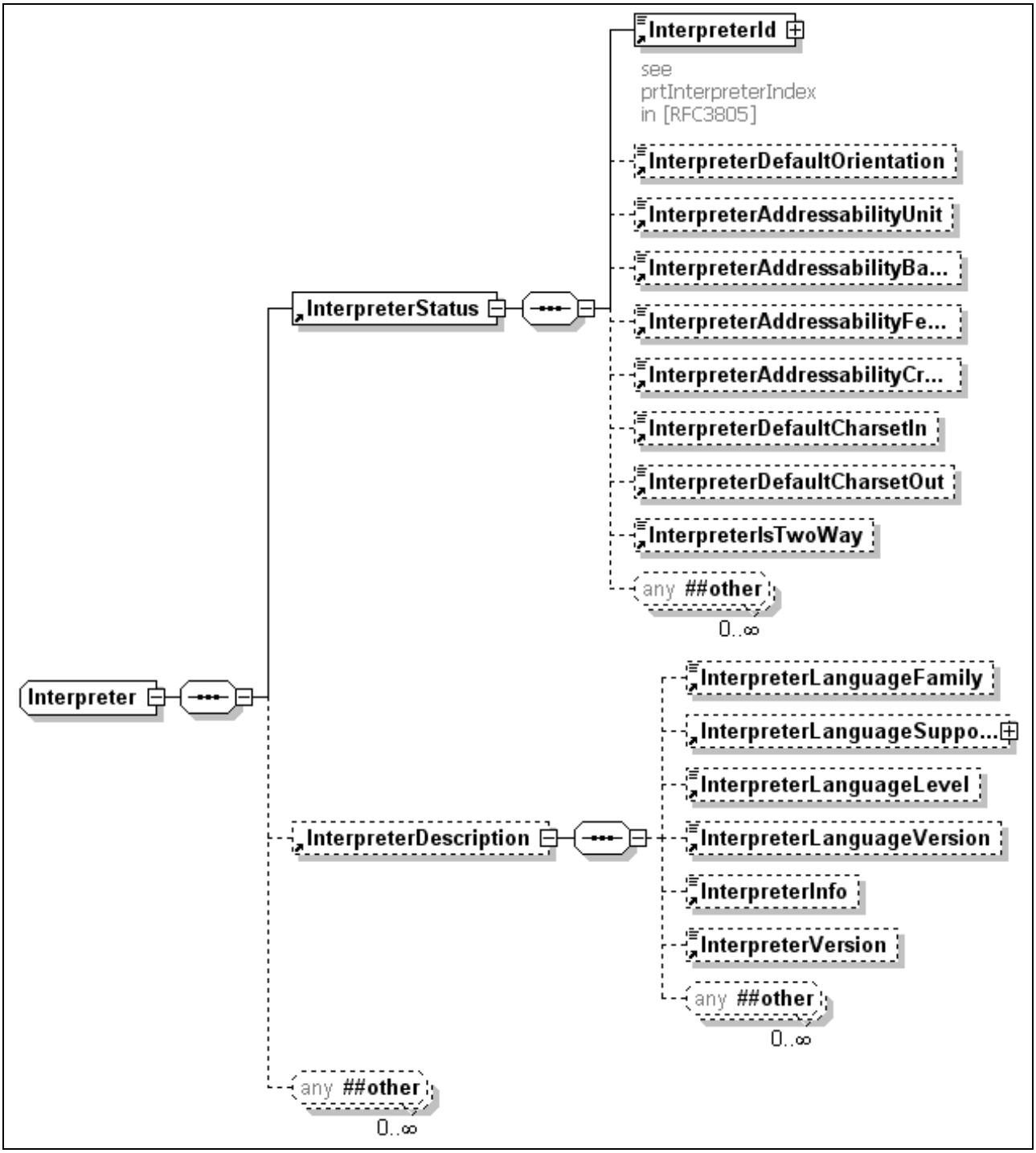


Figure 19 Interpreter

7.1.4.7.1 InterpreterID

(int) This element uniquely identifies an instance of an Interpreter. See [RFC3805] for additional details on prtInterpreterIndex.

7.1.4.7.2 InterpreterDefaultOrientation

(keyword) The current orientation default for this interpreter. (*Keywords: Other, Unknown, Portrait, Landscape*) See [RFC3805] prtInterpreterDefaultOrientation for additional details.

7.1.4.7.3 InterpreterAddressabilityUnit

(keyword) The units used for Interpreter Addressability. (*Keywords: Other, Unknown, TenThousandthsOfInches, Micrometers*) See [RFC3805] prtMarkerAddressabilityUnits for additional details.

7.1.4.7.4 InterpreterAddressabilityBasis

(keyword) This element expresses the confidence level of the addressability information for the Interpreter. *(Keywords: Other, Unknown, NotEmpty, Actual*)

7.1.4.7.5 InterpreterAddressabilityCrossFeed

(int) This element expresses the addressability in the cross feed direction (i.e. fast scan direction) for the Interpreter. This is expressed in InterpreterAddressabilityUnits. See [RFC3805] prtInterpreterXFeedAddressability for additional details.

7.1.4.7.6 InterpreterAddressabilityFeed

(int) This element expresses the addressability in the feed direction (i.e. fast slow direction) for the Interpreter. This is expressed in InterpreterAddressabilityUnits. See [RFC3805] prtInterpreterFeedAddressability for additional details.

7.1.4.7.7 InterpreterDefaultCharsetIn

(string) The default coded character set for input octets encountered outside a context in which the Page Description Language established the interpretation of the octets. See [RFC3805] prtInterpreterDefaultCharSetIn for additional details.

7.1.4.7.8 InterpreterDefaultCharsetOut

(string) The default character set for data coming from this interpreter through the output channel (i.e. the 'backchannel'). See [RFC3805] prtInterpreterDefaultCharSetOut for additional details.

7.1.4.7.9 InterpreterIsTwoWay

(boolean) Indicates whether or not this interpreter returns information back to the host. See [RFC3805] prtInterpreterTwoWay for additional details.

7.1.4.7.10 InterpreterLanguageFamily

(keyword) The family name of the control language which this interpreter is interpreting or emulating. The keyword values are extensible. (*Keywords include: Other, Unknown, LangPCL, LangTIFF, LangTIPSI*)See [RFC3805] prtInterpreterLangFamily for additional details.

7.1.4.7.11 InterpreterLanguageFamilySupported

(keyword) The family names of the control languages which this interpreter can interpret or emulate. The list of keyword is extensible. (*Keywords include: Other, Unknown, LangPCL, LangTIFF, LangTIPSI*)See [RFC3805] prtInterpreterLangFamily for additional details.

7.1.4.7.12 InterpreterLanguageLevel

(string) The level of the language which this interpreter is interpreting or emulating. This might contain a value like '5e'for an interpreter which is emulating level 5e of the PCL language. See [RFC3805 prtInterpreterLangLevel] for additional details.

7.1.4.7.13 InterpreterLanguageVersion

(string) The date code or version of the language which this interpreter is interpreting or emulating. See [RFC3805] prtInterpreterLangVersion for additional details.

7.1.4.7.14 InterpreterInfo

(string) A localized string to identify this interpreter. It is anticipated that this string will allow manufacturers to unambiguously identify their interpreters. See [RFC3805] o prtInterpreterDescription for additional details.

7.1.4.7.15 InterpreterVersion

(string) The date code, version number, or other product specific information tied to this interpreter. This value is associated with the interpreter, rather than with the version of the language which is being interpreted or emulated. See [RFC3805] prtInterpreterVersion for additional details.

7.1.4.8 Marker

(complex – Marker) Not applicable to Scan Service. See [RFC3805] prtMarkerTable for additional details.

7.1.4.9 MediaPaths

(complex – MediaPath) Not applicable to Scan Service. See [RFC3805] prtMediaPathTable for additional details.See ScanMediaPaths below.

7.1.4.10 Output Channel

(complex – OutputChannel) This element represents the destination pathways for Digital Documents. Output Channels are associated with an underlying Interface and are usually bound to a transport protocol or file system. See [RFC3805] prtChannelTable for additional details.

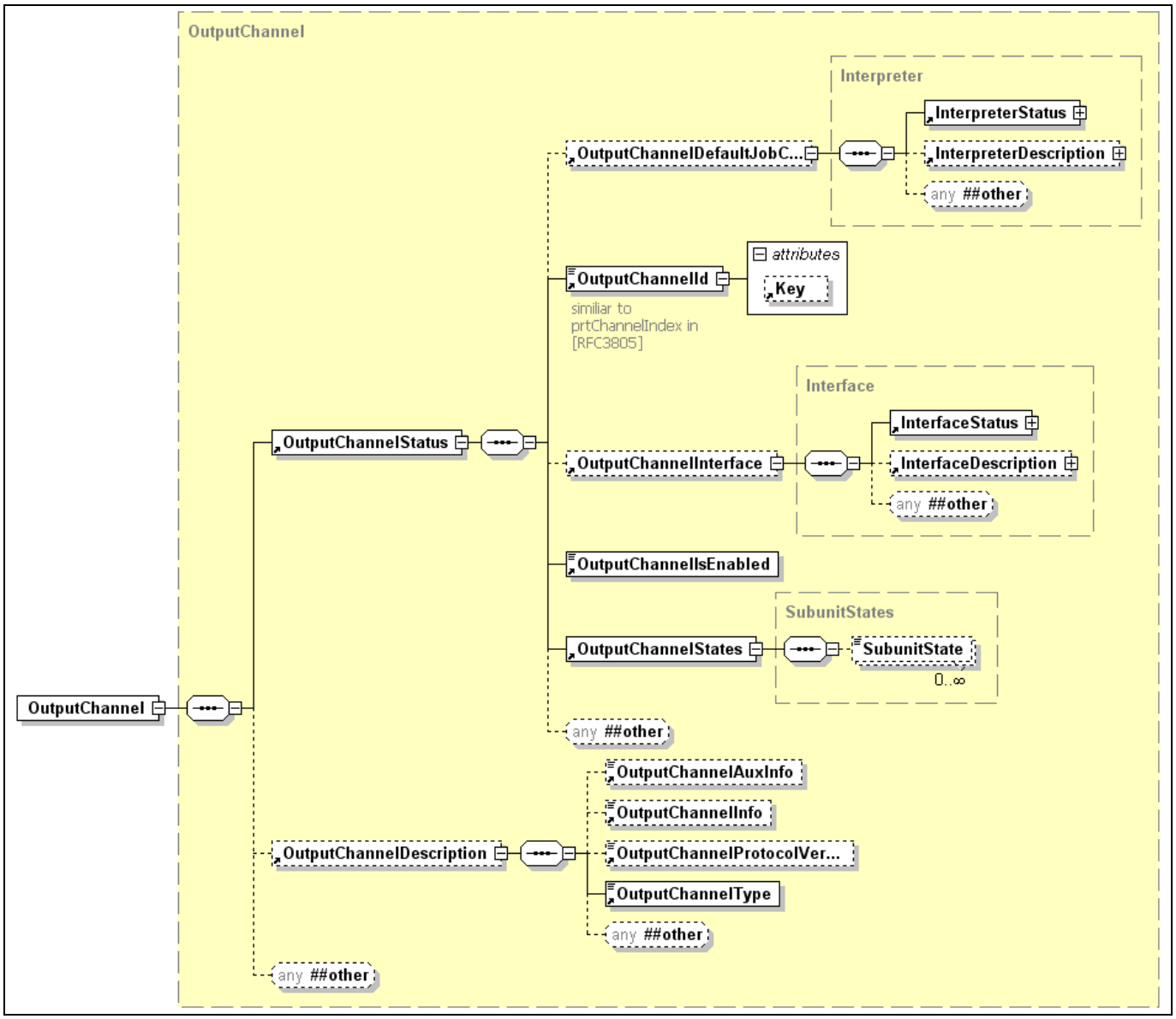


Figure 20 Output Channel

7.1.4.10.1 OutputChannelDefaultJobControlLanguage

(complex) This associated Job Control Language (i.e. Interpreter) defines the syntax used for control functions (e.g. queries, setting Document boundaries). See §7.1.4.6.1 See [RFC3805] prtChannelDefaultPageDescLangIndex for additional details.

7.1.4.10.2 OutputChannelID

(int) This element uniquely identifies an instance of an Output Channel. See [RFC3805] prtChannelIndex for additional details.

7.1.4.10.3 OutputChannelInterface

(complex) This is the Interface instance associated with this instance of an Output Channel. See §7.1.4.6 See [RFC3805] prtChannelIfIndex for additional details on Interface.

7.1.4.10.4 OutputChannelIsEnabled

(boolean) This element indicates whether or not an Output Channel instance is available for use.

7.1.4.10.5 OutputChannelStates

(list of keyword) This element shows the current States of the Output Channel. The States for the Output Channel are taken from the Input Channel (i.e. prtChannel) of the Printer MIB. See [RFC3805] prtChannelStatus for additional details.

7.1.4.10.6 OutputChannelAuxInfo

(string) This element provides additional protocol information needed to use the Output Channel's protocol. The information is protocol specific.

7.1.4.10.7 OutputChannelInfo

(string) This element provides protocol information needed to use the Output Channel's protocol. The information is protocol specific. See [RFC3805] prtChannelInformation for additional details.

7.1.4.10.8 OutputChannelProtocolVersion

(string) This element specifies the version of the protocol associated with an Output Channel instance in a protocol specific manner. See [RFC3805] prtChannelProtocolVersion for additional details .

7.1.4.10.9 OutputChannelType

(keyword) This element specifies the protocol associated with an Output Channel instance. See [RFC3805] prtChannelType for additional details.

7.1.4.10.10 Any

This element provides an extension point for vendor differentiation and implementation specific extensions while maintaining interoperability.

7.1.4.11 OutputTrays

(complex – OutputTray) This element represents the general purpose output trays capable of receiving Hard Copy Documents that have been scanned. Note: if the scanner is equipped with an ADF this element may not be applicable. If the ADF is the only destination of the Hard Copy Documents handled by the ADF (i.e. recirculating) then a general purpose Output Tray is not applicable. See ScanMediaPath for additional information.

See [RFC3805] for details on the prtOutputTable elements.

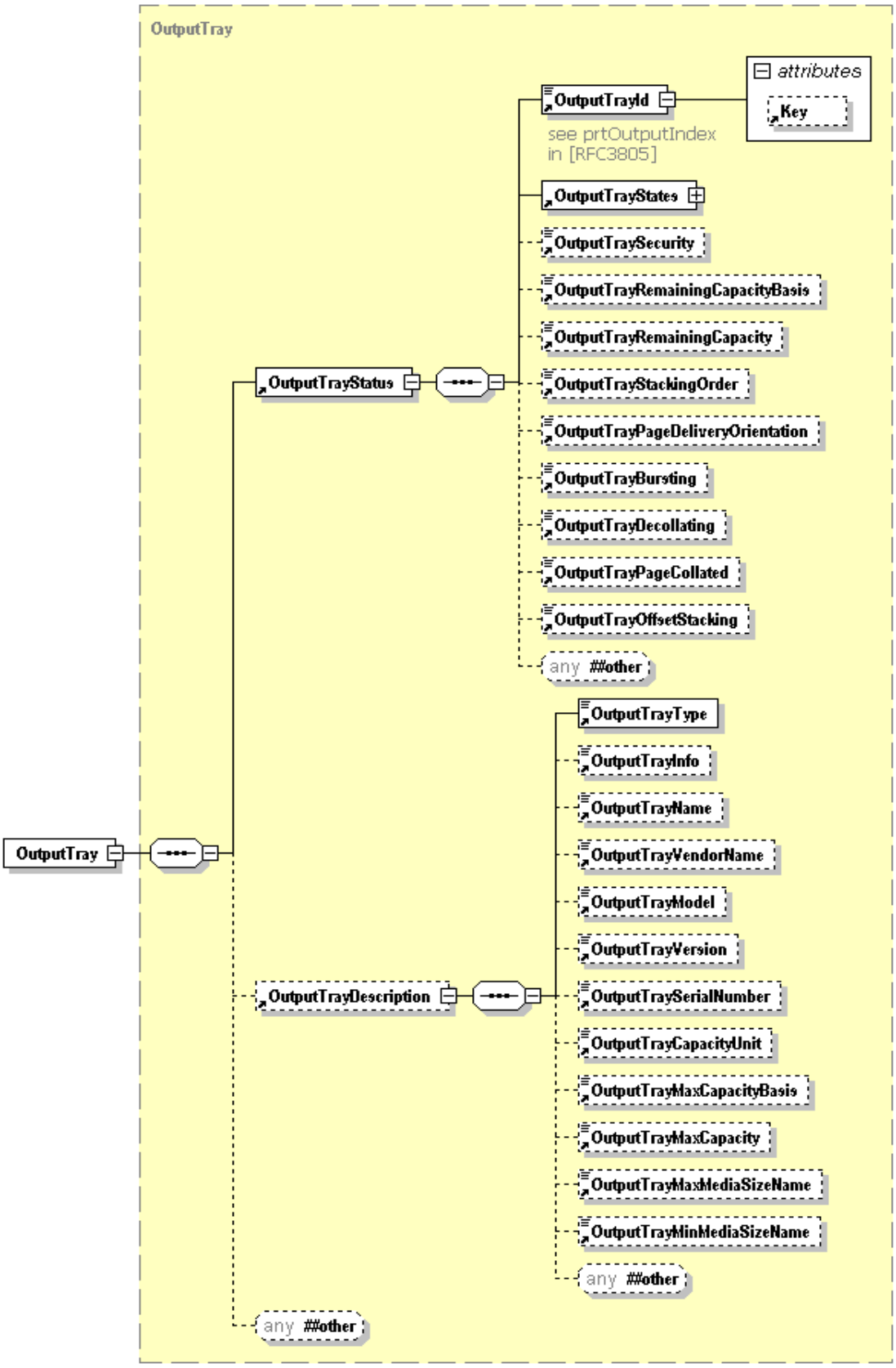


Figure 21 Output Tray

7.1.4.12 ScanMediaPath

(complex – ScanMediaPath) This element represents the media handling aspects of a Scanner.

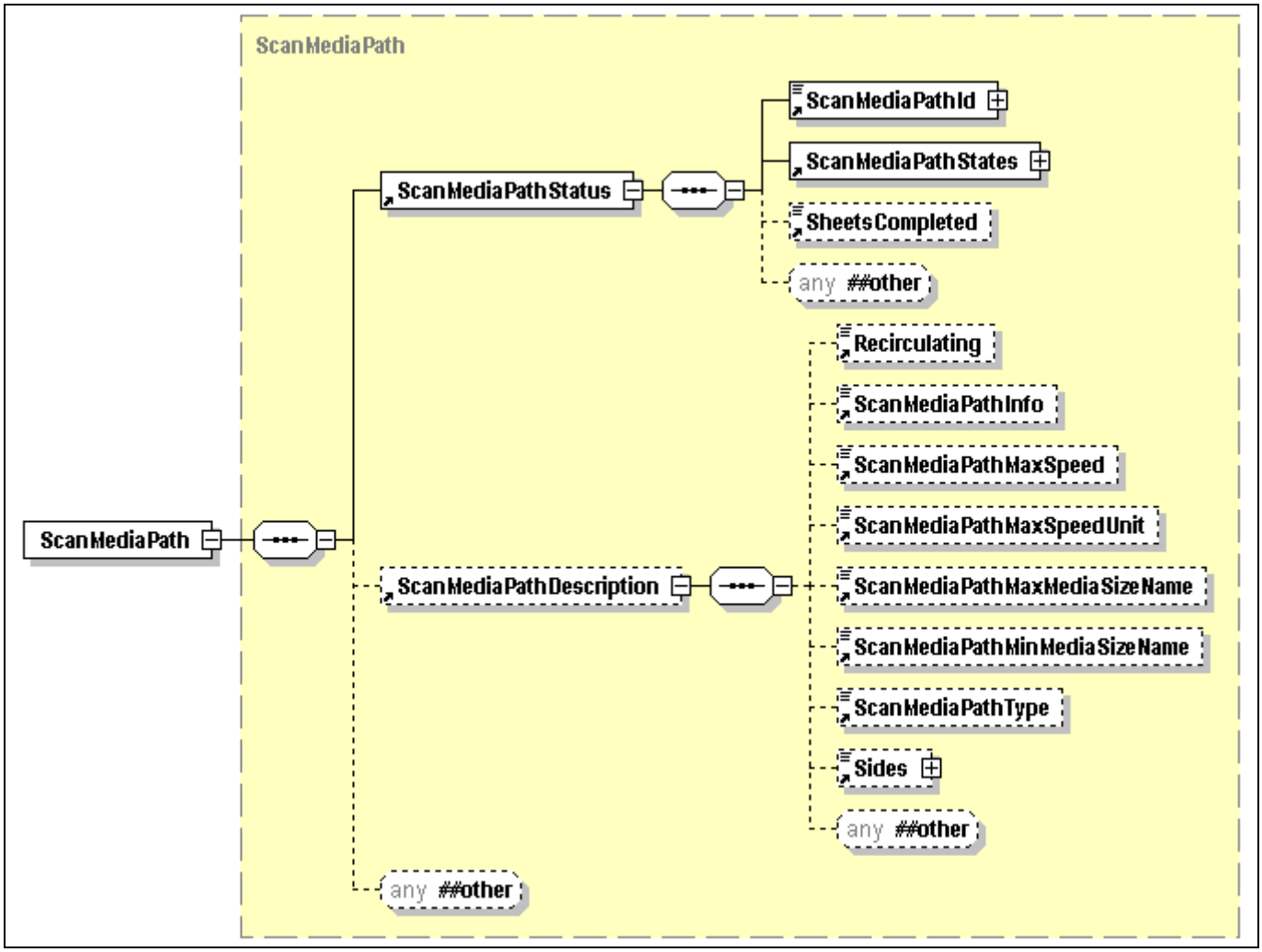


Figure 22 ScanMediaPath

7.1.4.12.1 ScanMediaPathId

(int) This element uniquely identifies an instance of a Scan Media Path.

7.1.4.12.2 ScanMediaPathStates

(list of keyword) This element shows the current States of the Scan Media Path. The States for the Scan Media Path are taken from the Media Path (i.e. prtMediaPathStatus) of the Printer MIB. See [RFC3805] prtMediaPathStatus for additional details.

7.1.4.12.3 SheetsCompleted

(int) This element represents the number of times a media sheets has passed through the ScanMediaPath.

7.1.4.12.4 Recirculating

(boolean) This element is applicable to Scan Media Paths that are capable of feeding Hard Copy Documents. If this element is set to the value ‘true’ then the Scan Media Path is both an input and output for the Hard Copy

Documents as they are to the Scanner Subunit. If the element is missing or the value is set to 'false' then the Hard Copy Document is delivered to an Output Bin after scanning is complete.

7.1.4.12.5 ScanMediaPathInfo

(string) This element provides additional information about the Scan Media Path instance. See [RFC3805] prtMediaPathDescription for additional details.

7.1.4.12.6 ScanMediaPathMaxSpeed

(int) This element indicates the maximum speed of Scan Media Path in ScanMediaPathMaxSpeedUnit. See [RFC3805] prtMediaPathMaxSpeed for additional details.

7.1.4.12.7 ScanMediaPathMaxSpeedUnit

(keyword) This element indicates the units used for the maximum speed of Scan Media Path. (*Keywords: Other, Unknown, TenThousandthsOfInchesPerHour, MicrometersPerHour, CharactersPerHour, LinesPerHour, ImpressionsPerHour, SheetsPerHour, DotRowPerHour, FeetPerHour, MetersPerHour*) See [RFC3805] prtMediaPathMaxSpeedPrintUnit for additional details.

7.1.4.12.8 ScanMediaPathMaxMediaSizeName

(keyword) This element indicates the maximum size of media that can flow through the Scan Media Path in Media Self Describing Name keywords (e.g. na_letter_8.5x11in, iso_a4_210x297mm). [PWG5101.1]

7.1.4.12.9 ScanMediaPathType

(keyword) This element indicates the type of Scan Media Path. (*Keywords: Other, Unknown, ADF, FilmReader, Platen*) See [RFC3805] prtMediaPathType for additional details.

7.1.4.12.10 Sides

(keyword)This element indicates the ability of the ScanMediaPath to scan multiple sides of an input media sheet. (*Keywords: OneSided, TwoSidedLongEdge, TwoSidedShortEdge*) See [RFC2911] sides for additional details.

7.1.4.13 Scanners

(complex – Scanner) This element represents the image acquisition aspects of a Scanner (i.e. Scanner Subunit).

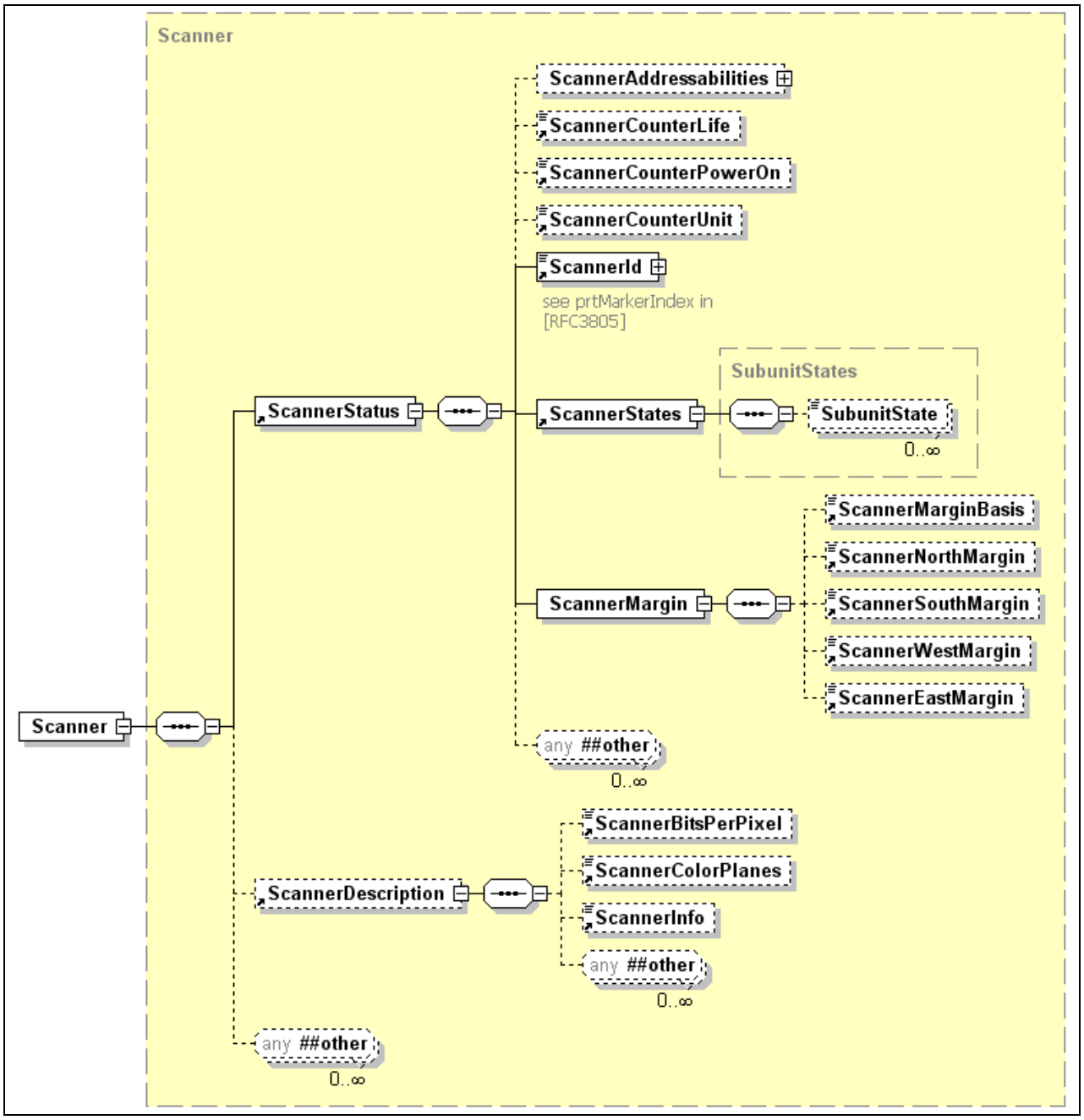


Figure 23 Scanner Subunit

7.1.4.13.1 ScannerAddressabilities

(complex) This element group contains information on the available addressabilities (i.e. resolution) of the Scanning hardware. See [RFC3805] prtMarkerAddressability objects for additional information.

7.1.4.13.1.1 ScannerAddressabilityBasis

(keyword) This element expresses the confidence level of the addressability information for the Scanning hardware. *(Keywords: Other, Unknown, NotEmpty, Actual*)

7.1.4.13.1.2 ScannerAddressabilityCrossFeed

(int) This element expresses the resolution in the cross feed direction (i.e. fast scan direction) for the Scanning hardware. This is expressed in ScannerAddressabilityUnits. See [RFC3805] prtInterpreterXFeedAddressability for additional details.

7.1.4.13.1.3 ScannerAddressabilityFeed

(int) This element expresses the resolution in the feed direction (i.e. slow scan direction) for the Scanning hardware. This is expressed in ScannerAddressabilityUnits. See [RFC3805] prtInterpreterFeedAddressability for additional details.

7.1.4.13.1.4 ScannerAddressabilityUnit

(keyword) This element expresses the units used to express the resolution of the feed directions. (*Keywords: Other, Unknown, TenThousanthsOfInches, Micrometers*)

7.1.4.13.2 ScannerCounterLife

(int) This element records the number of scans performed by the scanning hardware for the life of the device. The units are measured in ScannerCounterUnits. See [RFC3805] prtMarkerLifeCount objects for additional information.

7.1.4.13.3 ScannerCounterPowerOn

(int) This element records the number of scans performed by the scanning hardware since the device was lasted turned on. The units are measured in ScannerCounterUnits. See [RFC3805] prtMarkerPowerOnCount objects for additional information.

7.1.4.13.4 ScannerCounterUnits

(keyword) This element specifies units for ScannerCounterLife and ScannerCounterPowerOn. See [RFC3805] §6 prtMarkerCounterUnit objects for additional information. (*Keywords: Other, Unknown, TenThousanthsOfInches, Micrometers, Characters, Lines, Impressions, Sheets, DotRow, Hours, Feet, Meters*)

7.1.4.13.5 ScannerId

(int) This element uniquely identifies an instance of a image acquisition subunit (i.e. Scanner subunit).

7.1.4.13.6 ScannerStates

(list of keyword) This element shows the current States of the Scanner. The States for the Scanner are taken from the Marker (i.e. prtMarkerStatus) of the Printer MIB. See [RFC3805] §6 for additional details on prtMarkerStatus. (*Keywords: AvailableAndIdle, AvailableAndStandby, AvailableAndActive, AvailableAndBusy, UnavailableAndOnRequest, UnavailableAndBroken, Unknown, WarningAlerts, CriticalAlerts, Online, Offline, Transitioning*)

7.1.4.13.7 ScannerMargin

(complex) This element group contains information on the Margin of the Scanning hardware. The margins provide a bounding box within which the Scanner hardware is capable of scanning. See [RFC3805] prtMarkerMargin objects for additional information.

7.1.4.13.7.1 ScannerMarginBasis

(keyword)This element expresses the confidence level of the Margin information for the Scanning hardware. *(Keywords: Other, Unknown, NotEmpty, Actual*)

7.1.4.13.7.2 ScannerNorthMargin

(int) This element expresses the offset from the leading edge of the scan hardware or platen where Image acquisition can occur. (See §6.5.1) The units are given in ScannerAddressabilityUnit.

7.1.4.13.7.3 ScannerSouthhMargin

(int) This element expresses the offset from the trailing edge of the scan hardware or platen where Image acquisition can occur. (See §6.5.1) The units are given in ScannerAddressabilityUnit.

7.1.4.13.7.4 ScannerWestMargin

(int) This element expresses the offset from the west edge of the scan hardware or platen where Image acquisition can occur. (See §6.5.1) The units are given in ScannerAddressabilityUnit.

7.1.4.13.7.5 ScannerEastMargin

(int) This element expresses the offset from the east edge of the scan hardware or platen where Image acquisition can occur. (See §6.5.1) The units are given in ScannerAddressabilityUnit.

7.1.4.13.8 ScannerBitsPerPixel

(int) This element contains the number of bits per pixel used by the scanner's image acquisition subsystem.

7.1.4.13.9 ScannerColorPlanes

(int) This element contains the number of color planes used by the scanner's image acquisition subsystem.

7.1.4.13.10 ScannerInfo

(string) This element contains descriptive information about the scanner's image acquisition subsystem.

7.1.4.14 Storage

(complex – Storage) Not applicable to a Scan Service. Note that the Scan Service specification pushes data to a ScanDestination. Although the hosting device may contain internal storage access to data stored on that storage is not available through the Scan Service itself.

7.1.4.15 VendorSubunits

This element provides extension points for vendor and implementation specific Subunits while maintaining interoperability.

7.1.5 Scan Service Description

(complex – ScanServiceDescription) Below is a view of the Description elements for the Scan Service. The Description elements provide Descriptive information for the Scan Service. The elements are administratively set. The element values can be modified directly or indirectly through an operation.

Note that ScanServiceDescription consists of two sequences. The first represents elements inherited from the Imaging Service super class and the second sequence includes Scan Service specific extensions.

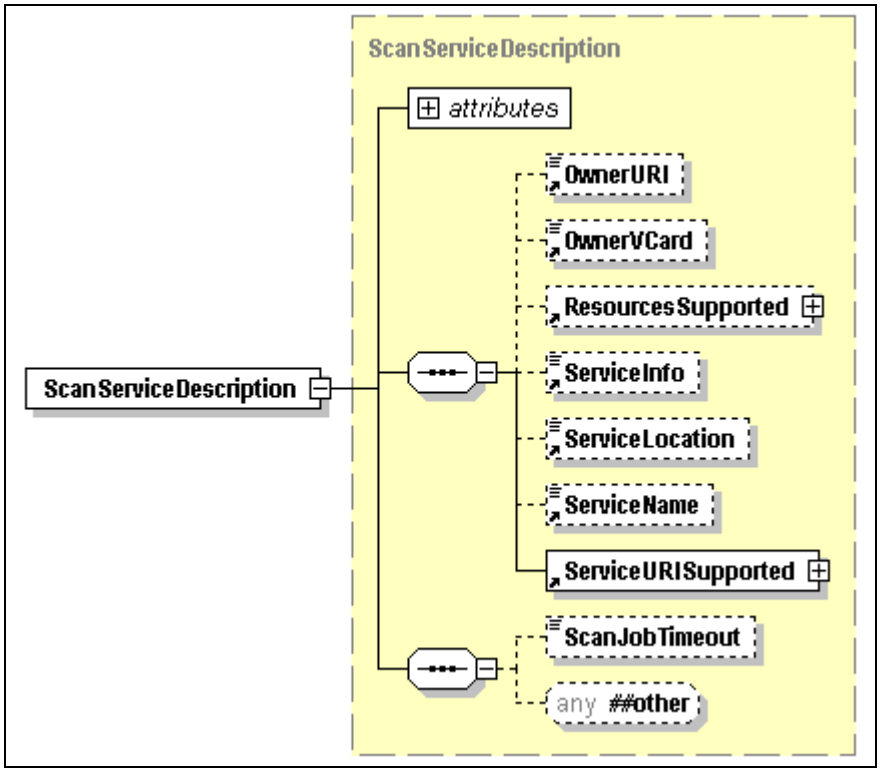


Figure 24 ScanServiceDescription

7.1.5.1 OwnerURI

(string) This is the URI by which you can reach the admin or owner who created the service.

7.1.5.2 OwnerVCard

(string) This is the vCard See [RFC2426] of the owner of the Scan Service.

7.1.5.3 ResourcesSupported

(list of complex) This is a service specific view of the resources used by the service.

7.1.5.4 ServiceInfo

(string) This is descriptive information about this Scan Service. (Example: “Out of courtesy for others, please scan only small (1-5 page) jobs at this scanner")

7.1.5.5 ServiceLocation

(string) This describes the physical location of this Scan Service. (Example: “Building 128 Floor 2 Room 210C")

7.1.5.6 ServiceName

(string) This is the end-user friendly name of Scan Service.

7.1.5.7 ServiceURISupported

(list of URI) This element contains the URIs where the Scan Service is available.

7.1.5.8 ScanJobTimeout

(int) When Scan Service is operating in MultipleSetOriginals mode, this element sets the timeout value for awaiting the End User's next set of Document originals. At timeout the Scan Service SHALL close the job inputs and schedule the job for processing.

7.1.5.9 Any

This element provides extension points for vendor differentiation and implementation specific extensions while maintaining interoperability.

7.1.6 Scan Service Status

(complex – ScanServiceStatus) Below is a view of the Status elements for the Scan Service. The Status elements provide state information for the Scan Service. The elements are maintained by automata and can not be directly set. The element values can be modified indirectly through an operation. For example PauseScanService operation on the Scan Service may result in the change of the State and StateReasons elements.

Note that ScanServiceStatus consists of two sequences. The first represents elements inherited from the Imaging Service super class and includes elements such as Id and State. The second sequence includes Scan Service specific extensions to the super class such as the Scan Service counters.

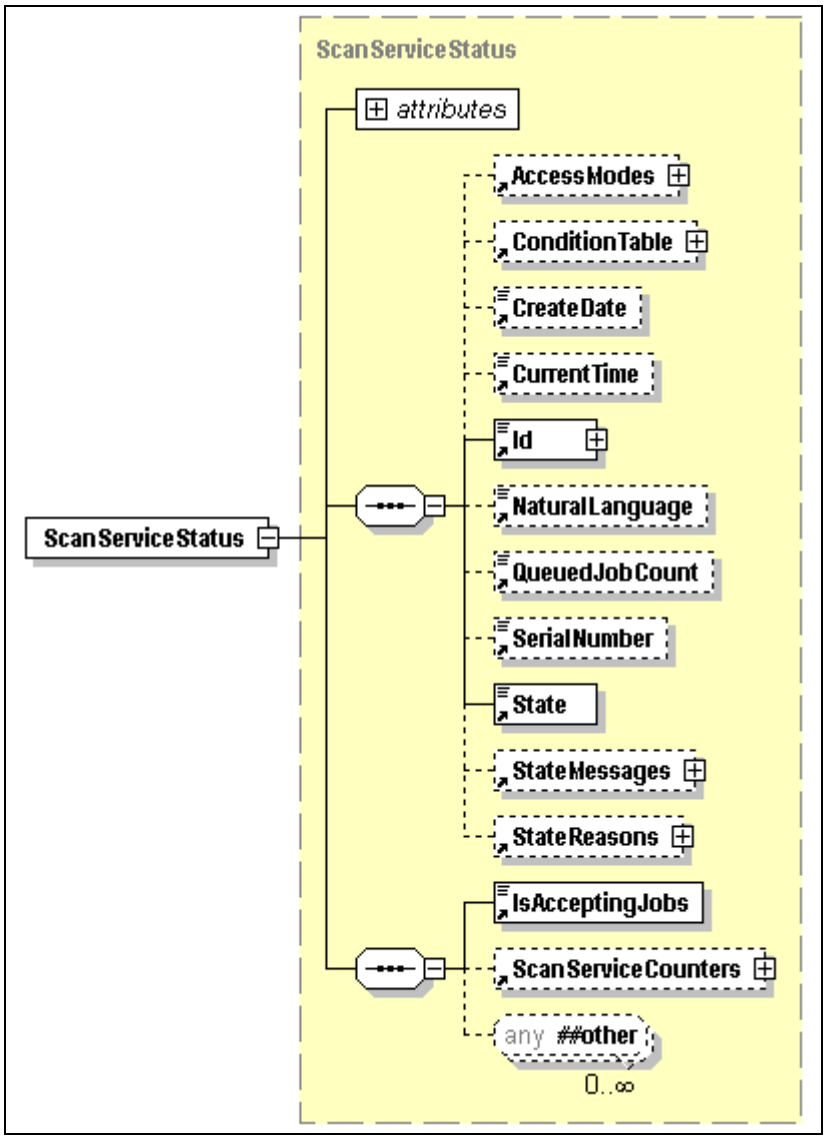


Figure 25 Scan Service Status

7.1.6.1 AccessModes

(list of keyword)This element corresponds to the access mode property of a POSIX file and specifies the basic access control policy for the Scan Service object, as set by the Owner. The AccessMode element takes precedence over any externally specified access policy.

The AccessMode element describes the basic access control policy for the Owner, members of the Owner's group (e.g., Administrators), and all other authenticated users (i.e., the 'World'). Typically, only the Owner or a member of the Owner's group (site-specific) is granted the Write permission, but the World is granted Read permission (e.g., to read elements such as capabilities) and Execute permission (e.g., to send Job creation operations to the Scan Service object).[CHMOD, STAT] (*Keywords: OwnerRead, OwnerWrite, OwnerExecute, GroupRead, GroupWrite, GroupExecute, WorldRead, WorldWrite, WorldExecute*)

7.1.6.2 Condition Table

The Condition Table provides additional information on the current and past state of Services and Subunits. See §7.2 below.

7.1.6.3 CreateDate

(dateTime) This element indicates date and time that the Service was created.

7.1.6.4 CurrentTime

(dateTime) This element indicates the current date and time according the Scan Services internal clock.

7.1.6.5 ID

(int) This element indicates a 32 bit Object Identifier for the Scan Service instance. It is unique within a scan server.

7.1.6.6 IsAcceptingJobs

(boolean) This element indicates whether this Scan Service is currently able to accept jobs. How the Value for this element is configured is implementation-specific, e.g., local console, web page.

7.1.6.7 NaturalLanguage

(string) This element indicates the local language currently used by the Service. This is the language used unless the operation specifies a RequestedNaturalLanguage(example values: en-us, fr, de, ja) See [rfc3066].

7.1.6.8 QueuedJobCount

(int) This element specifies the number of jobs this Scan Service has currently accepted but not yet finished.

7.1.6.9 SerialNumber

(string) This element specifies the serial number of the device hosting the service.

7.1.6.10 State

(keyword) This element identifies the current state of scanning service. The state is a unification of the service states from IPP and the Host Resource MIB. See [RFC2911] and [RFC2790]. The state transition diagram represents the IPP view. The state values MUST NOT be extended by an implementation.

Values:

From rfc2911:

Idle – The Scan Service is available and can start processing a new job.
Processing – The Scan Service is currently processing jobs.
Stopped - No jobs can be processed and intervention is needed.

From rfc2790:

Unknown– The state of the Scan Service is not known.
Testing- The Scan Service is in testing or maintenance mode.

Down- The Scan Service is unavailable for jobs.

7.1.6.10.1 Service State Diagram

Service State Diagram is divided into three phases:
- <Init> - Unknown state - immediately after service creation
- <Offline> - Down and Testing states - no user jobs are processed
- <Online> - Idle, Processing, and Stopped states

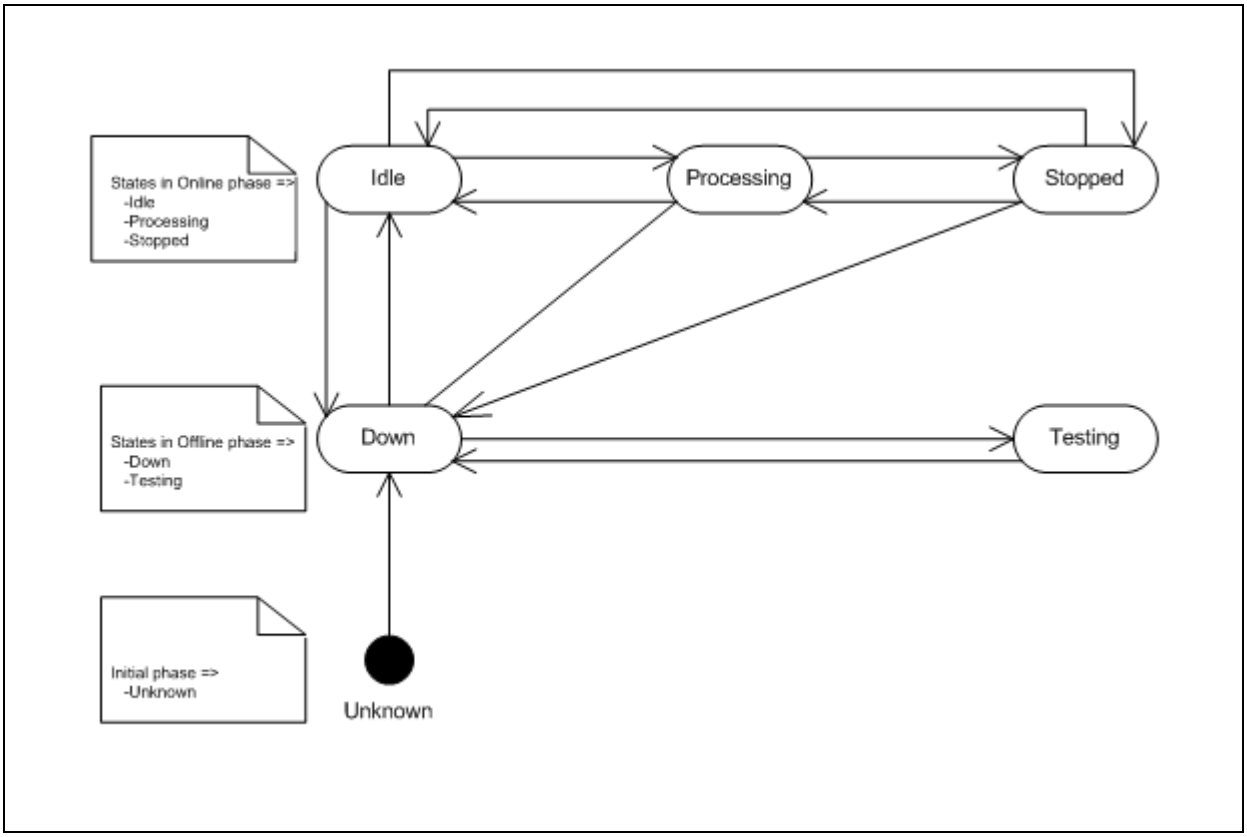


Figure 26 Top Level Service State Transition Diagram

7.1.6.10.2 Service State Transition Tables

The following notations are used in the two subsequent service state transition tables:

~ = logical NOT (e.g., ~C.paused means "not paused")
C = prefix of a condition
E = prefix of an event (e.g., E.endJob means "job completed")
titlecase = state (e.g., Idle), operation (e.g., Startup), or phase
lowercase = action function (in FSM)
error = In Table 2 indicates that an error response is returned to the requestor
In Table 3 indicates that an error message should be logged indicating an implementation error

The following notes are used in the two service state transition tables:

(1) Startup and Restart
- Startup (Unknown/Init) sends E.startup and goes to (Down/Offline)
- Startup (Down/Offline) is a synonym for Restart
- Restart (Down/Offline) initializes and goes to (Idle/Online)

(2) Shutdown
- Shutdown (Testing|Idle|Stopped) goes to (Down/Offline)
- Shutdown (Processing) sends E.shutdown and stays in (Processing)

(3) Testing
- No Test related operation or events are defined is this specification or protocol and they are included to indicate the transition is made in an implementation specific manner

General note: C.isAcceptingJobs represents the ability of the service to create new Jobs (i.e. CreateScanJob will fail if ~C.isAcceptingJobs). Other operations are not affected by this condition.

7.1.6.10.2.1 Service State Transition By Operations

SERVICE STATE MACHINE (Operations)					
Input	**State**				
	Down	**Testing**	**Idle**	**Processing**	**Stopped**
Operation (Condition)	**Action (new state)**	**Action (new state)**	**Action (new state)**	**Action (new state)**	**Action (new state)**
DisableScanService	error	disable (~C.IsAcceptingJobs)	disable (~C.IsAcceptingJobs)	disable (~C.IsAcceptingJobs)	disable (~C.IsAcceptingJobs)
EnableScanService	error	enable (C.IsAcceptingJobs)	enable (C.IsAcceptingJobs)	enable (C.IsAcceptingJobs)	enable (C.IsAcceptingJobs)
PauseScanService	error	pause (C.Pause)	pause (Stopped, C.Pause)	pause (Stopped, C.Pause)	pause (C. Pause)
ResumeScanService	error	resume (~C.Pause)	resume (~C.Pause)	resume (~C.Pause)	resume (Idle, ~C.Pause)
RestartScanService (Note 1)	restart (Idle)	restart (Idle)	restart (Idle)	restart (Idle)	restart (Idle)
ShutdownScanService (Note 2)	error	shutdown (Down)	shutdown (Down)	shutdown (Down)	shutdown (Down)
StartupScanService (Note 1)	restart (Idle)	error	error	error	error
test (Note 3)	test (Testing)	test (Testing)	error	error	error

Table 2 Service State Machine (Operations)

7.1.6.10.2.2 Service State Transition By Events

Note that the following table should be read from top to bottom. Consecutive entries of the same event are differentiated by differing associated conditions. The last entry in the series is equivalent to the event with condition other than the ones already covered.

	SERVICE STATE MACHINE (Events)
Input	**State**

	Down	Testing	Idle	Processing	Stopped
Event (Condition)	**Event or Condition (new state)**	**Event or Condition (new state)**	**Event or Condition (new state)**	**Event or Condition (new state)**	**Event or Condition (new state)**
E.critical	C.critical	C.critical	C.critical (Stopped)	C.critical (Stopped)	C.critical
E.criticalCleared (i.e. last C.Critical)	~C.critical	~C.critical	error	error	~C.critical (Idle or Processing)
E.endJob (C.shutdown)	Error Add note for internal state error	shutdown (Down)	error	shutdown (Down)	error
E.endJob (C.paused)	error	C.paused (Testing)	error	C.paused (Stopped)	error
E.endJob	error	(Testing)	error	schedule (Idle or Processing)	error
E. Warning	C.warning	C.warning	C.warning	C.warning	C.warning
E. warningCleared (i.e. last C.warning)	~C.warning	~C.warning	~C.warning	~C.warning	~C.warning
E.startup (Note 1)	restart (Idle)	error	error	error	error
E.startJob (C.paused)	error	error (Testing)	error	error	error
E.startJob	error	schedule (Testing)	schedule (Processing)	schedule	error
E.testing (Note 3) E.testingCleared	(Testing) error	(Testing) (Down)	error error	error error	error error

Table 3 Service State Machine (Events)

7.1.6.10.3 Detailed Service State Transition Diagram

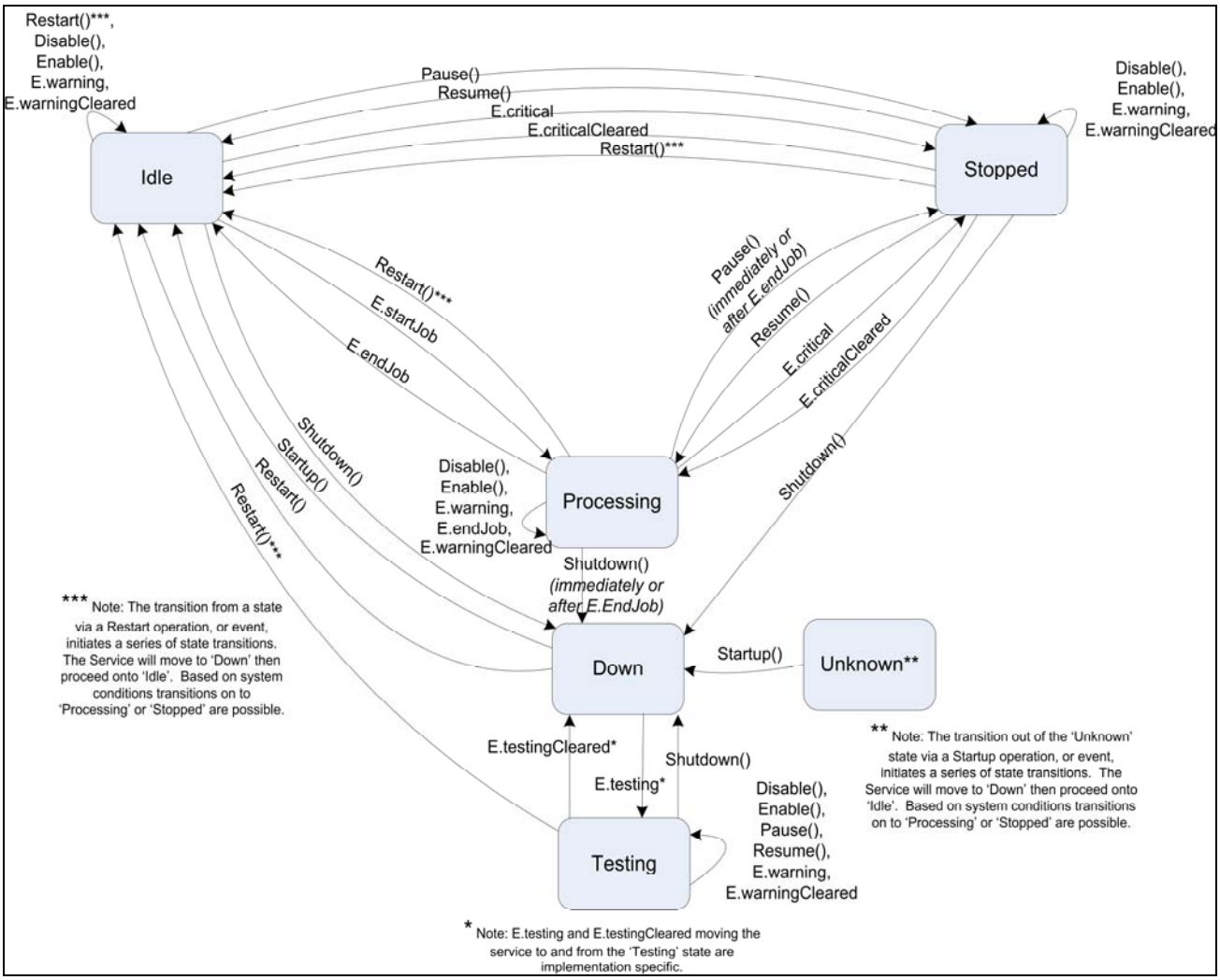


Figure 27 Detailed Service Transition Diagram

7.1.6.11 StateReasons

(list of keyword) This element supplies additional detail about the service's state. The keywords are extensible. The standard keyword values are defined in §4.4.12 of [rfc2911] and §4.4.3.1 of [WS-Scan]. (*Keywords: AttentionRequired, Calibrating , ConnectingToDevice, CoverClosed, CoverOpen, Deactivated, DoorOpen, HoldNewJobs, InputCannotFeedSizeSelected, InputManualInputRequest, InputTrayEmpty, InputTrayMissing, InputTrayPositionFailure, InterpreterResourceUnavailable, InterlockClosed, InterlockOpen, InternalStorageFull, LampError , LampWarming , MediaJam, MediaLow, MediaNeeded, MovingToPaused, MultipleFeedError, OutputAreaAlmostFull, OutputAreaFull, OutputTrayMissing, Paused, Shutdown, Spool-areaFull, Stopping, StoppedPartly, TimedOut, SubunitAdded, SubunitAlmostEmpty, SubunitAlmostFull, SubunitAtLimit, SubunitClosed, SubunitEmpty, SubunitFull, SubunitLifeAlmostOver, SubunitLifeOver, SubunitMemoryExhausted, SubunitMissing, SubunitMotorFailure, SubunitNearLimit, SubunitOffline, SubunitOpened, SubunitOverTemperature, SubunitPowerSaver, SubunitRecoverableFailure, SubunitRecoverableStorageError, SubunitRemoved, SubunitResourceAdded, SubunitResourceRemoved, SubunitThermistorFailure, SubunitTimingFailure, SubunitTurnedOff, SubunitTurnedOn, SubunitUnderTemperature, SubunitUnrecoverableFailure, SubunitUnrecoverableStorageError, SubunitWarmingUp, AlertRemovalOfBinaryChangeEntry,*

ConfigurationChange, InputTrayElevationFailure, InterpreterCartridgeAdded, InterpreterCartridgeDeleted, InterpreterComplexPageEncountered, InterpreterMemoryDecrease, InterpreterMemoryIncrease, InterpreterResourceAdded, InterpreterResourceDeleted, MediaPathCannotDuplexMediaSelected, MediaPathMediaTrayAlmostFull, MediaPathMediaTrayFull, MediaPathMediaTrayMissing, OutputMailboxSelectFailure, PowerDown, PowerUp, None, Other, Unknown)

7.1.6.12 StateMessages

(list of string) This element contains information about the Service State and StateReasons in human readable text. If the Service supports this element, it MUST be able to generate the messages in any of the natural languages supported by the Service.

7.1.6.13 ScanServiceCounters

(complex) This element contains the counters for the amount of work performed by the Scan Service, timers covering utilization and monitoring information covering errors, warnings, traffic, job counts and configuration changes. See [PWG5106.1-2007].

7.1.6.14 Any

This element provides an extension point for vendor differentiation and implementation specific extensions while maintaining interoperability.

7.2 Condition Table

(complex – ConditionTable) The Condition Table provides additional information on the current and past state of Services and Subunits. The contents of the tables are based on the Alert Table from the Printer MIB.

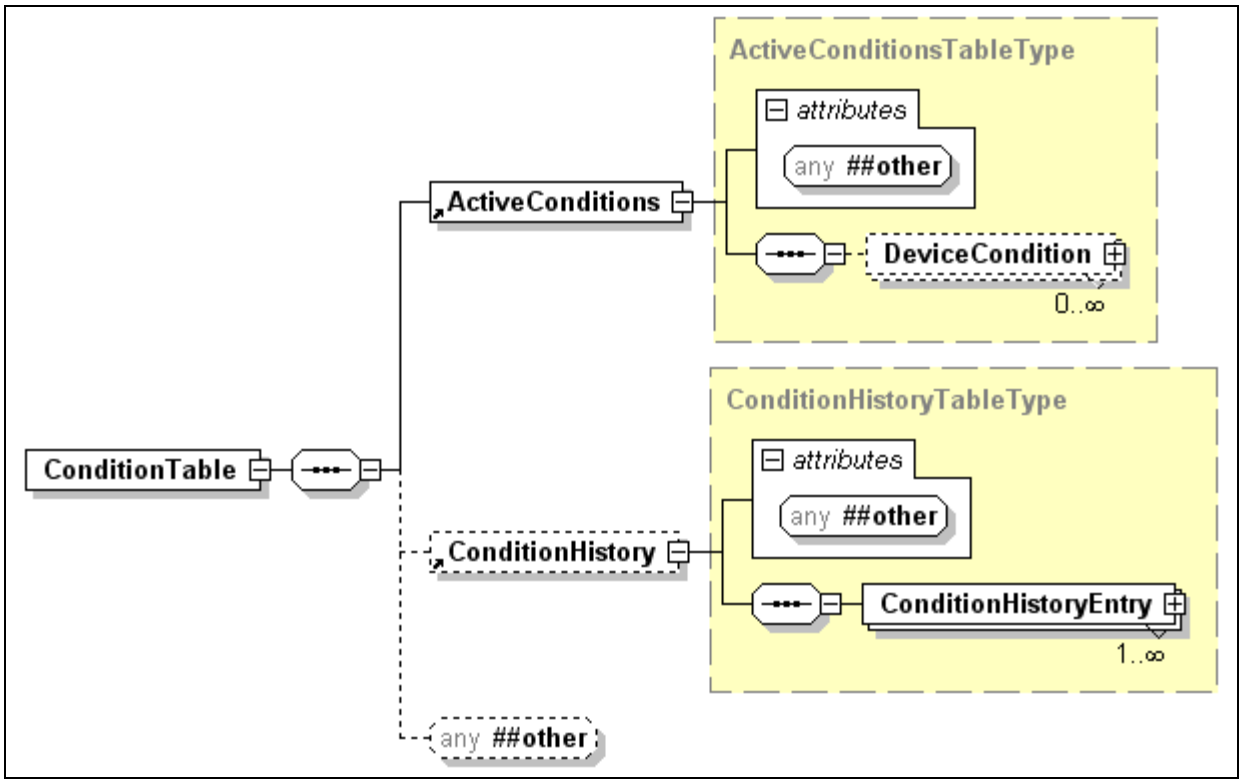


Figure 28 Condition Table

7.2.1 ActiveConditions

(list of complex – DeviceCondition) This table represents the conditions (informational, warning or critical) that are currently in effect on the device.

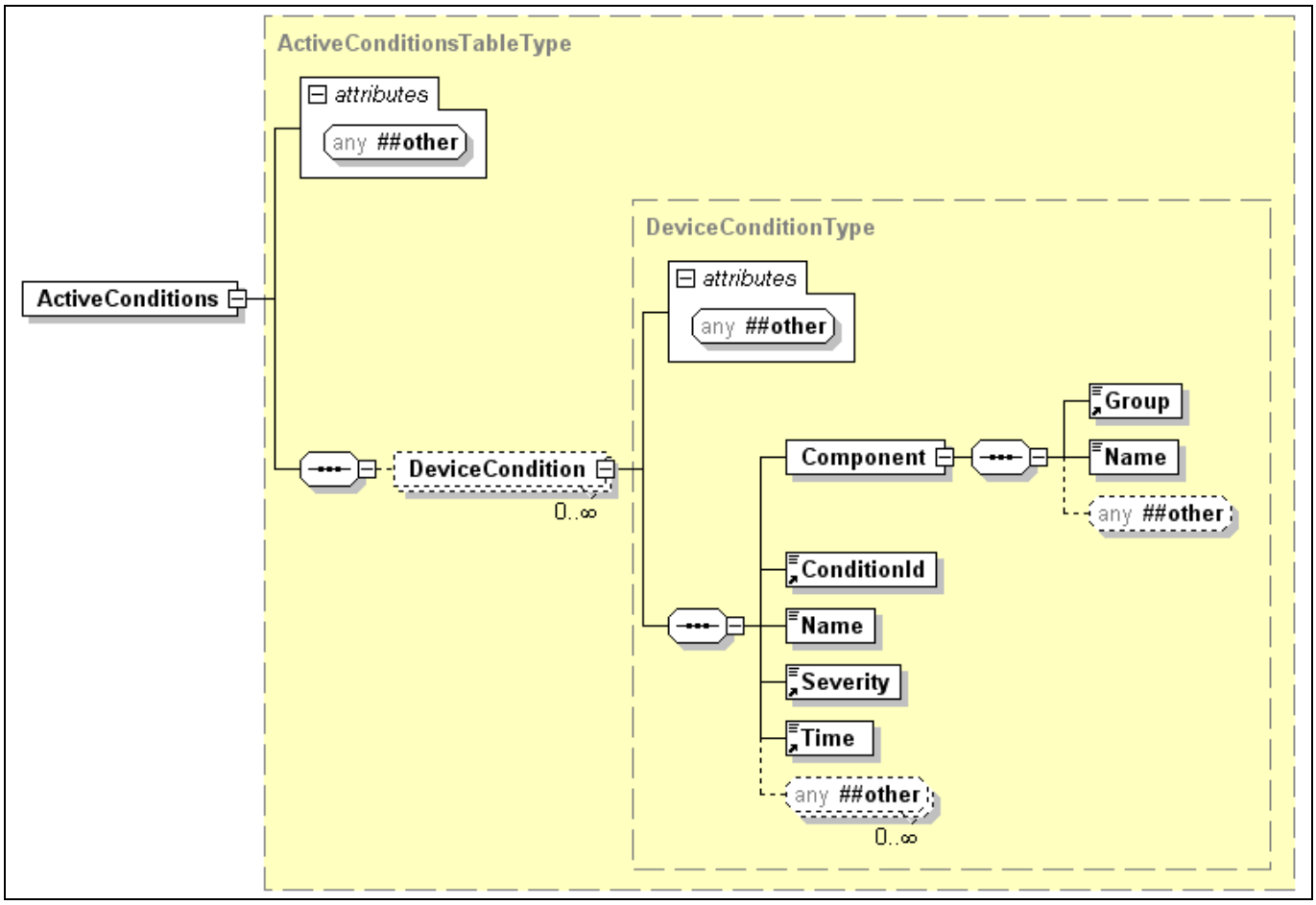


Figure 29 ActiveConditions

7.2.1.1 Component

(complex) This element specifies the origin of the condition.

7.2.1.1.1 Group

(keyword) This element specifies the class of the component (e.g. subunit, service) that is the source of the condition. (*Keywords: Console, Finisher, InputChannel, InputTray, Interface, Interpreter, Marker, MediaPath, OutputChannel, OutputTray, ScanMediaPath, Scanner, VendorSubunit, CopyService, EmailInService, EmailOutService, FaxInService, FaxOutService, NetFaxInService, NetFaxOutService, PrintService, ScanService, TemplateService, TransformService*)

7.2.1.1.2 Name

(string) This element specifies the instance of the component (e.g. subunit, service) that is the source of the condition. Values are taken from the Name element of the Group.

7.2.1.2 ConditionId

(int)This element is the unique identifier for the condition.

7.2.1.3 Name

(keyword) This element is the keyword that identifies the condition type. The values are the same values that are used for Service's StateReasons and are extensible. (*Keywords: AttentionRequired, Calibrating , ConnectingToDevice, CoverClosed, CoverOpen, Deactivated, DeveloperEmpty, DeveloperLow, DoorOpen, FuserOverTemp, FuserUnderTemp, HoldNewJobs, InputCannotFeedSizeSelected, InputManualInputRequest, InputMediaChangeRequest, InputTrayEmpty, InputTrayMissing, InputTrayPositionFailure, InterpreterResourceUnavailable, InterlockClosed, InterlockOpen, InternalStorageFull, LampError , LampWarming , MarkerFailure, MarkerSupplyEmpty, MarkerSupplyLow, MarkerWasteAlmostFull, MarkerWasteFull, MediaEmpty, MediaJam, MediaLow, MediaNeeded, MovingToPaused, MultipleFeedError, None, OpcLifeOver, OpcNearEol, Other, OutputAreaAlmostFull, OutputAreaFull, OutputTrayMissing, Paused, Shutdown, Spool-areaFull, Stopping, StoppedPartly, TimedOut, TonerEmpty, TonerLow, Unknown*)

7.2.1.4 Severity

(keyword) This element is the keyword that identifies criticality of the condition. (*Keywords:* Informational, Warning, Critical)

7.2.1.5 Time

(dateTime) This element is the date and time the condition occurred.

7.2.2 ConditionsHistory

(list of complex – ConditionHistoryEntry) This table represents the conditions (informational, warning or critical) that have been cleared from the device. This element is an extension of the data type used by ActiveConditions. See above for the elements that are common. Only the ConditionHistory specific elements are explained below.

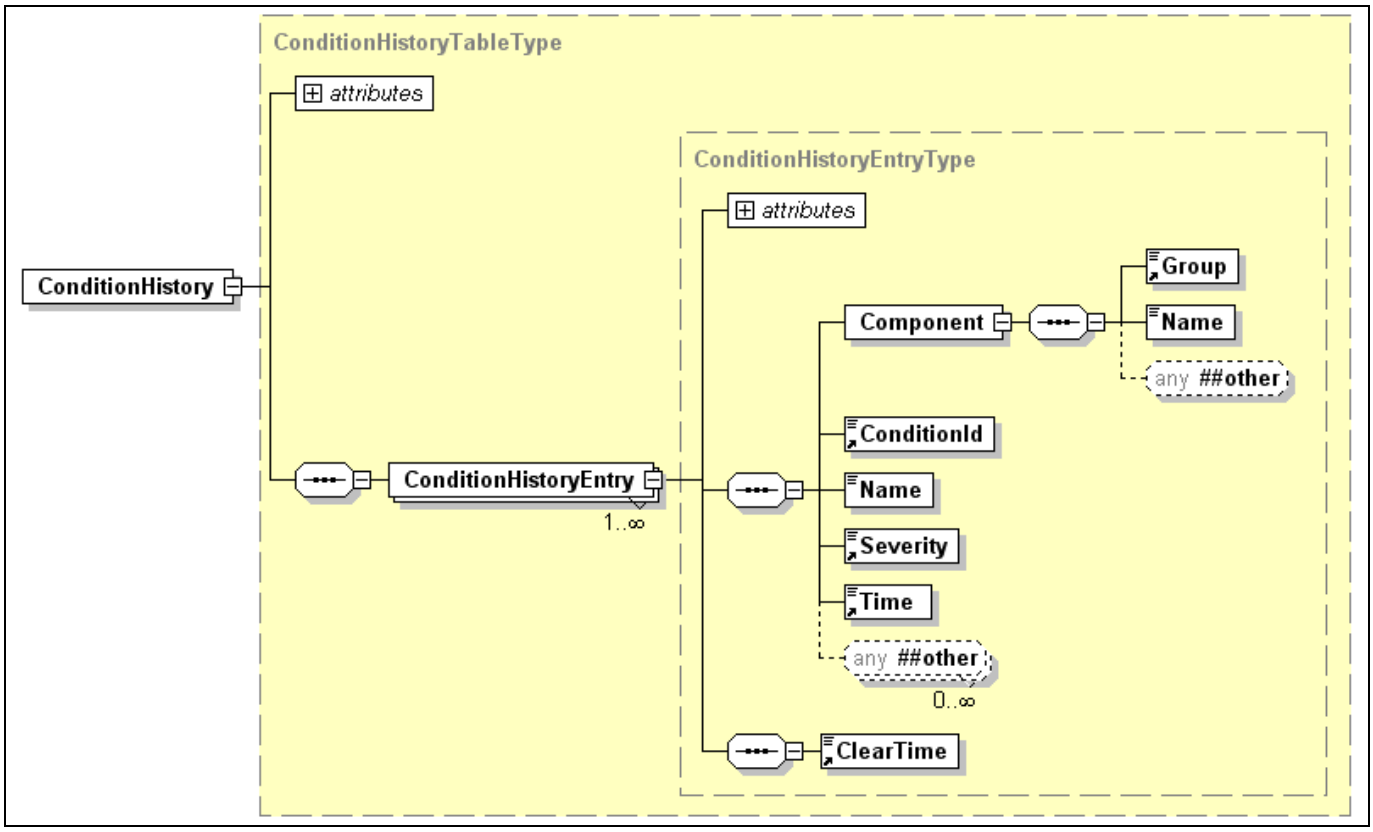


Figure 30 ConditionHistory

7.2.2.1 ClearTime

(dateTime) This element is the date and time the condition was removed from the ActiveConditions list.

8 Scan Job Model definition

(complex – JobTable) This section provides the detailed descriptions of the elements that comprise the Scan Job Table. Included is the JobTable that contains the lists of ActiveJobs and JobHistory as well as the ScanJob itself.

Below is the top level view of the Scan Job. 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, Canceled) appear in the JobHistory list. Note that the JobHistory list is optional and the amount of time a Job is retained in the JobHistory list is implementation specific.

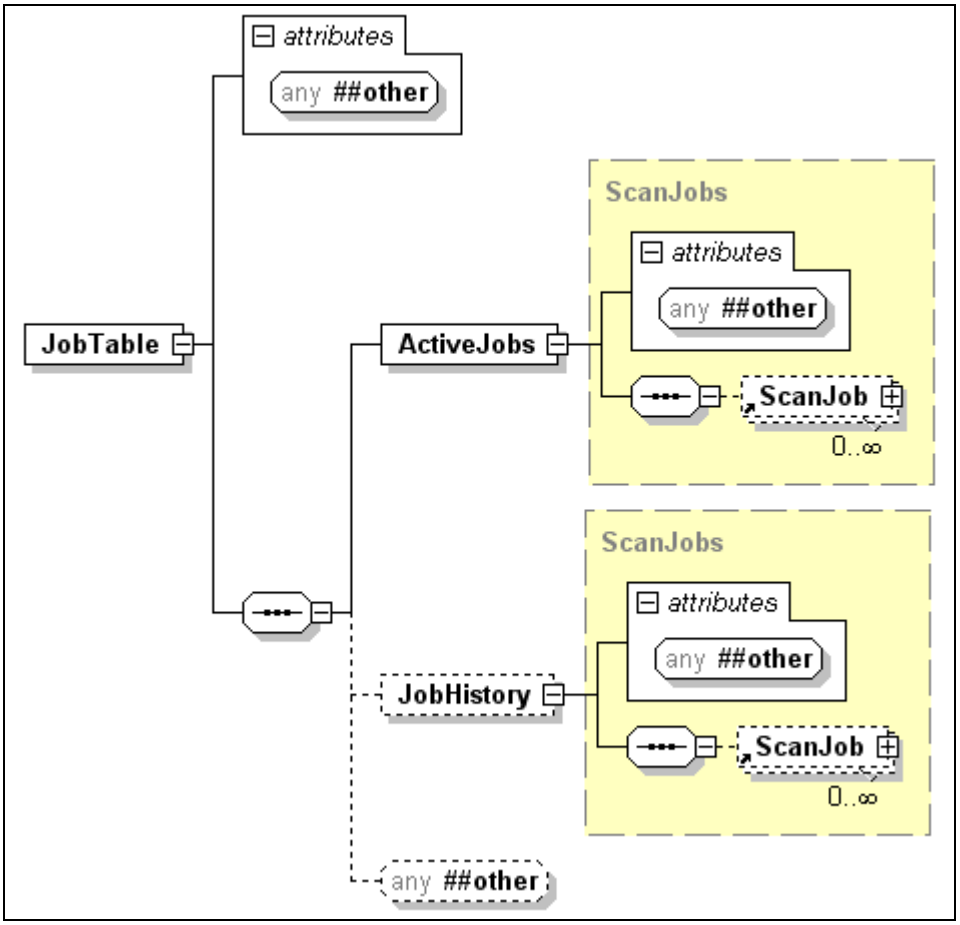


Figure 31 JobTable

Scan Jobs can contain 0 or more Documents. During job creation it is possible that temporarily there are 0 Documents. The state of the job is described in the ScanJobStatus element. ScanJobTicket 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 ScanDocumentTicket with the Scan Document. The ScanJobTicket and ScanDocumentTicket represent the End User's intent while the Job Receipt and Document Receipt represent what the Scan Service actually did.

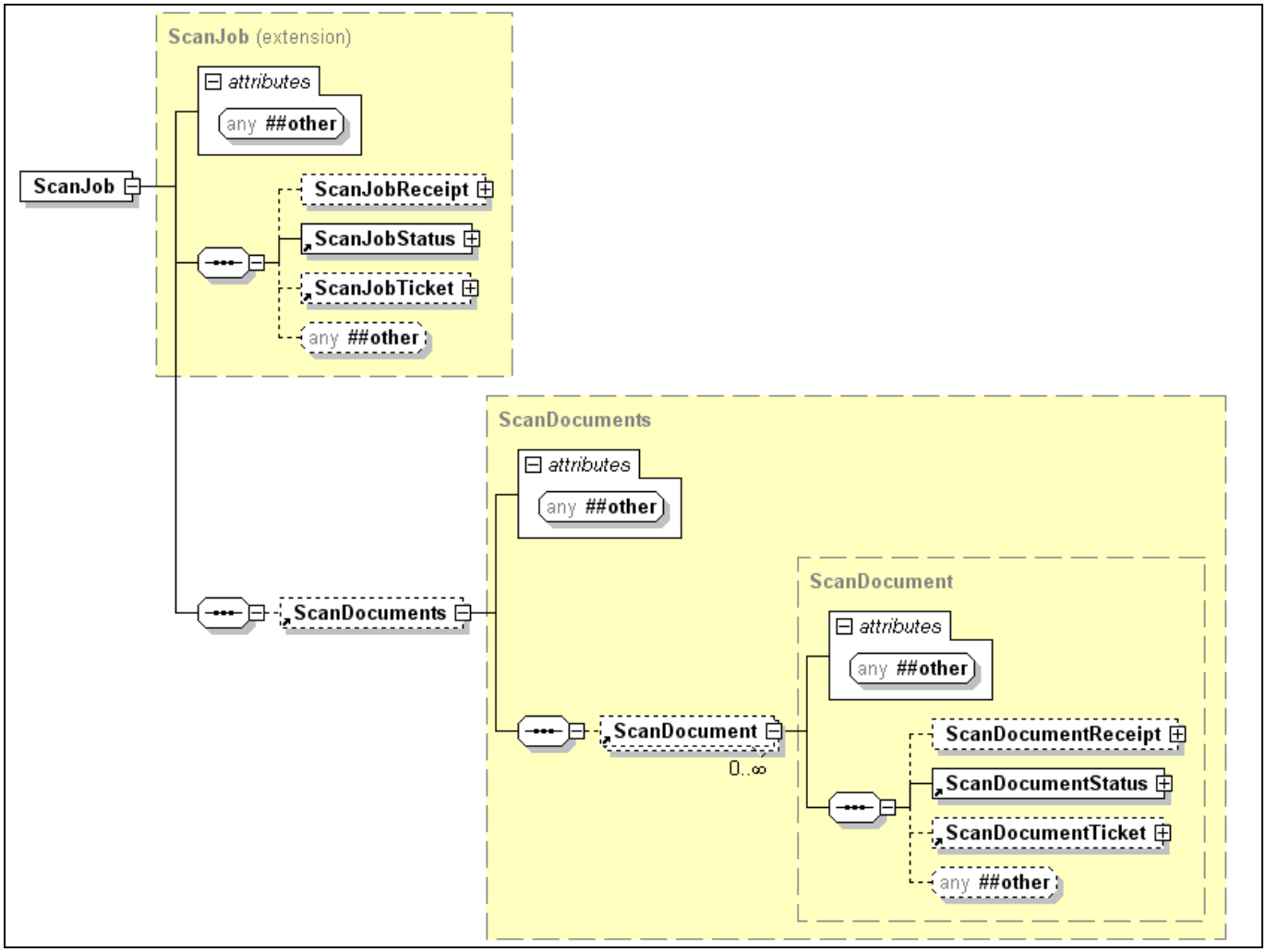


Figure 32 High Level Scan Job View

8.1.1 ScanJobReceipt

(complex - ScanJobReceipt) This element has exactly the same structure as the ScanJobTicket. For each processing element of a Scan Job, it records the actual value used by the Scan Service for processing the Scan Job. It contains the elements supplied by the Scan Client and applied to the job, any element or values substitutions made by the Scan Service and any default elements or values applied by the Scan Service. See §8.1.3 for element descriptions.

8.1.2 ScanJobStatus

(complex - ScanJobStatus) Below is a view of the Status elements for the Scan Job. The Status elements provide state information for the Scan Job. The elements are maintained by automata and can not be directly set. The element values can be modified indirectly through an operation. For example CancelScanJob operation on the Scan Job may result in the change of the State and StateReasons elements.

Note that ScanJobStatus consists of two sequences. The first represents elements inherited from the Imaging Job super class (i.e. ImagingJobStatus) and includes elements such as JobId and JobState. The second sequence includes Scan Service specific extensions to the super class such as the Scan Service counters.

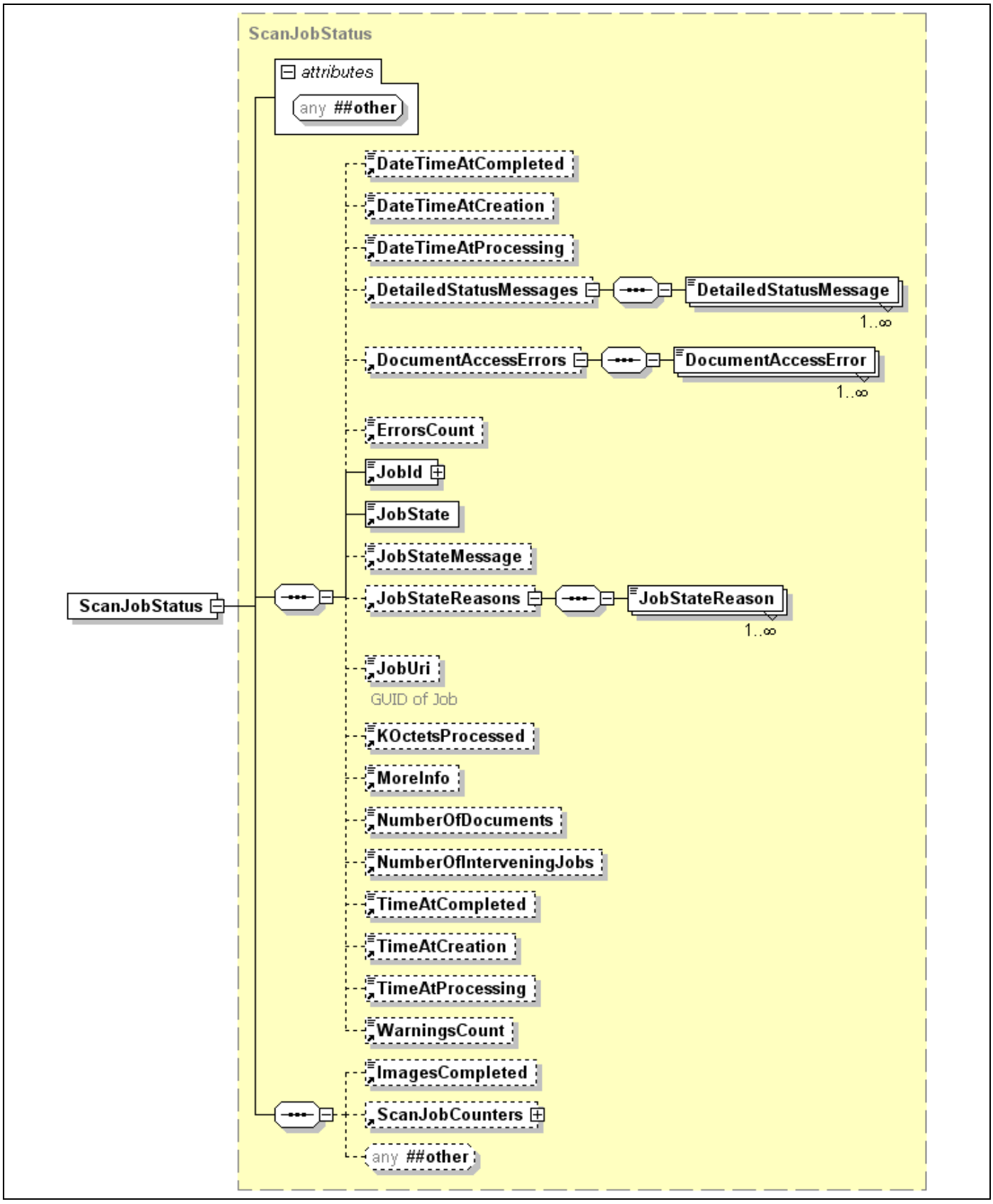


Figure 33 ScanJobStatus

8.1.2.1 DateTimeAtCompleted

(dateTime) This element indicates the date and time at which the Job object processing was completed (or was canceled or aborted).

8.1.2.2 DateTimeAtCreation

(dateTime) This element indicates the date and time at which the Job object was created.

8.1.2.3 DateTimeAtProcessing

(dateTime) This element indicates the date and time at which the Job object first began processing (i.e. scanned).

8.1.2.4 DetailedStatusMessages

(list of string) This element specifies additional detailed and technical information about the job. The Scan Service MAY localize the message(s), since they are intended for use by the system administrator or other experienced technical persons. Localization might obscure the technical meaning of such messages. Clients MUST NOT attempt to parse the value of this attribute.

8.1.2.5 DocumentAccessErrors

(list of string) This element provides additional information about each Document access error for this job encountered by the Scan Service after it returned a response to scan initiation operation and subsequently attempted to store the Image data to the locations supplied in the ScanJobTicket. For errors in the protocol associated with a URI scheme, such as 'http:' or 'ftp:', the error code is returned in parentheses, followed by the URI. For example:
(404) http://ftp.pwg.org/pub/pwg/mfd/job1207.png

8.1.2.6 ErrorsCount

(int) This element indicates the number of errors encountered while scanning the Hardcopy Document and generating and storing the Digital Document.

8.1.2.7 JobId

(int) JobId is an integer value that uniquely identifies the job within the Scan Service. The Scan Service is responsible for implementing a job numbering scheme that will not allow two different jobs to coexist with the same JobId.

8.1.2.8 JobState

(keyword) This element identifies the current state of Scan Job. The state values MUST NOT be extended by an implementation.
Values:

From rfc2911:

- **Aborted** – the job was halted due to an error
- **Canceled** – the job was halted due to a cancel operation
- **Completed** – the job has successfully completed operation
- **Pending** – the job has been accepted by the system and is awaiting system resources before it can start processing
- **PendingHeld** – the job is not a candidate for processing for any number of reasons and will return to the Pending state when the reasons are no longer present. The Job's JobStateReason MUST indicate the reason the Job is not a candidate for processing.
- **Processing** – the job is currently progressing towards completion. The job can be scanning, processing captured data or transferring Image data from the Scanner Subunit to the destination location.
- **ProcessingStopped** – the job has suspended processing and is awaiting resources to become available before processing can continue. Jobs in this state will be either awaiting image processing resources, internal storage or network availability for transferring Images.

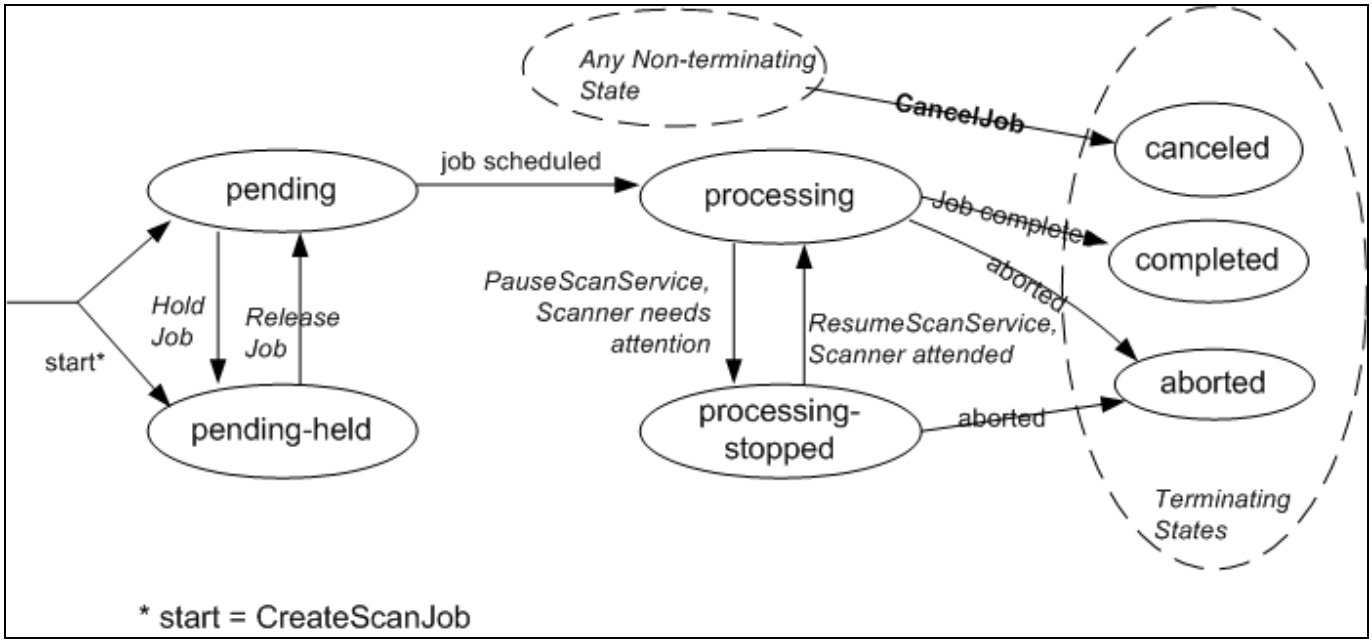


Figure 34 Job State Transition Diagram

8.1.2.9 JobStateMessages

(string) This element provides information about the Job State and StateReasons in human readable text. If the Service supports this element, it MUST be able to generate the messages in any of the natural languages supported by the Service.

8.1.2.10 JobStateReasons

(list of keyword) This element supplies additional detail about the Job state. The standard keyword values are defined in §4.3.8 of [RFC2911] and §4.5.1.3of [WS-Scan]. (*Keywords: AbortedBySystem, CompressionError, DigitalSignatureDidNotVerify, DigitalSignatureTypeNotSupported, DocumentAccessError, DocumentFormatError, ErrorsDetected, InvalidScanTicket, ImageTransferError, JobCanceledAtDevice, JobCanceledByOperator, JobCanceledByUser, JobCompletedSuccessfully, JobCompletedWithErrors, JobCompletedWithWarnings, JobDataInsufficient, JobDigitalSignatureWait, JobHoldUntilSpecified, JobIncoming, JobInterpreting, JobOutgoing, JobPasswordWait, JobQueued, JobResuming, JobScanning, JobScanningAndTransferring, JobScheduling, JobSpooling, JobStreaming, JobSuspended, JobSuspendedByOperator, JobSuspendedBySystem, JobSuspendedByUser, JobSuspending, JobTimedOut, JobTransferring , JobTransforming, JobWarningsDetected, None, ProcessingToStopPoint, QueuedInDevice, ResourcesAreNotReady, ResourcesAreNotSupported, ScannerStopped, ServiceOffLine, Stopped, SubmissionInterupted, UnsupportedCompression, UnsupportedDocumentFormat, WarningsDetected*)

8.1.2.11 JobUri

(string) This element contains the globally unique identifier for a job.

8.1.2.12 KOctetsProcessed

(int) This element indicates the total number of octets processed in integral units of 1024 octets so far.

8.1.2.13 MoreInfo

(string) This element is the URI used to obtain information intended for End User consumption about this specific Job.

8.1.2.14 NumberOfDocuments

(int) This element is the number of Documents in this Job.

8.1.2.15 NumberOfInterveningJobs

(int) This element is the number of jobs that are "ahead" of this Job assuming the current scheduled order.

8.1.2.16 TimeAtCompleted

(int) The time at which the Job completed in “ScannerUpTime” seconds.

8.1.2.17 TimeAtCreation

(int) This element is the time at which the Job was created in “ScannerUpTime” seconds.

8.1.2.18 TimeAtProcessing

(int) This element is the time at which the Job first began processing in “ScannerUpTime” seconds.

8.1.2.19 WarningsCount

(int) This element contains the total number of warnings that a Scan Service has generated while processing and delivering the Job’s Document(s).

8.1.2.20 ScanServiceCounters

(complex) This element contains the counters for the amount of work performed for this Document by the Scan Service, timers covering utilization and monitoring information covering errors, warnings , traffic, job counts are included. See [PWG5106.1-2007].

8.1.2.21 ImagesCompleted

(int) This element specifies the number of Images scanned. A sheet of media may be scanned multiple times. A single sheet of media may have multiple Regions scanned resulting in an increment of ImagesCompleted for each Region scanned. In two sided scanning each side of the sheet is scanned, generating two scans in the ImagesCompleted count. (Note: The scanner MUST update this element with the most exact information when it is available.) See [WS-Scan] § 4.5.1.6. (Note: the value for this element is the same as the value for the ScanServiceCounters.WorkTotals.Images element if implemented.)

8.1.2.22 Any

This element provides an extension point for vendor differentiation and implementation specific extensions while maintaining interoperability.

8.1.3 ScanJobTicket

(complex - ScanJobTicket) The ScanJobTicket contains description and processing elements provided by the Scan client during Scan Job creation. This information is used by the Scan Service during the processing of a Scan Job. This information is made available to Scan Clients through the GetScanJobElements operation and a subset is made available through the GetActiveScanJobs and GetScanJobHistory operations.

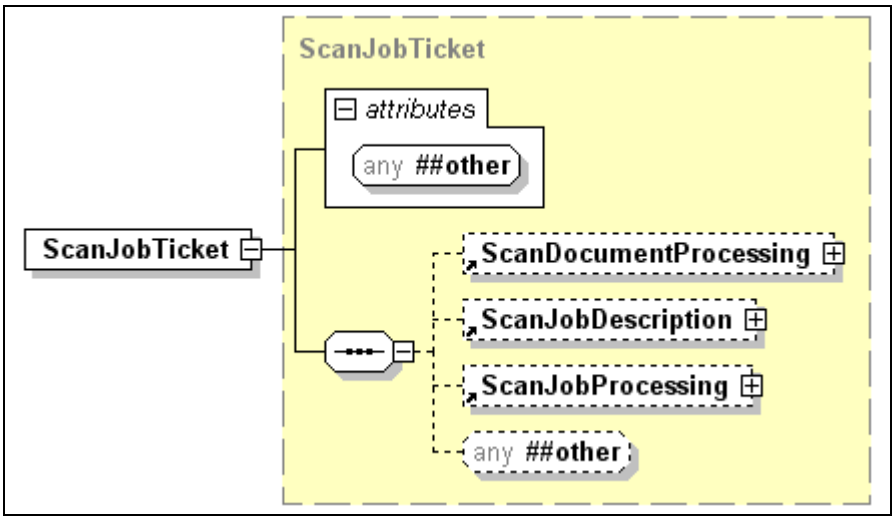


Figure 35 ScanJobTicket

8.1.3.1 Scan Document Processing

(complex - ScanDocumentProcessing) The ScanDocumentProcessing 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 user has requested.

Note that ScanDocumentCapabilities consists of two sequences. The first represents elements inherited from the Imaging Service super class and includes elements such as NumberUp. The second sequence includes Scan Service specific extensions to the super class such as AutoExposure and Brightness.

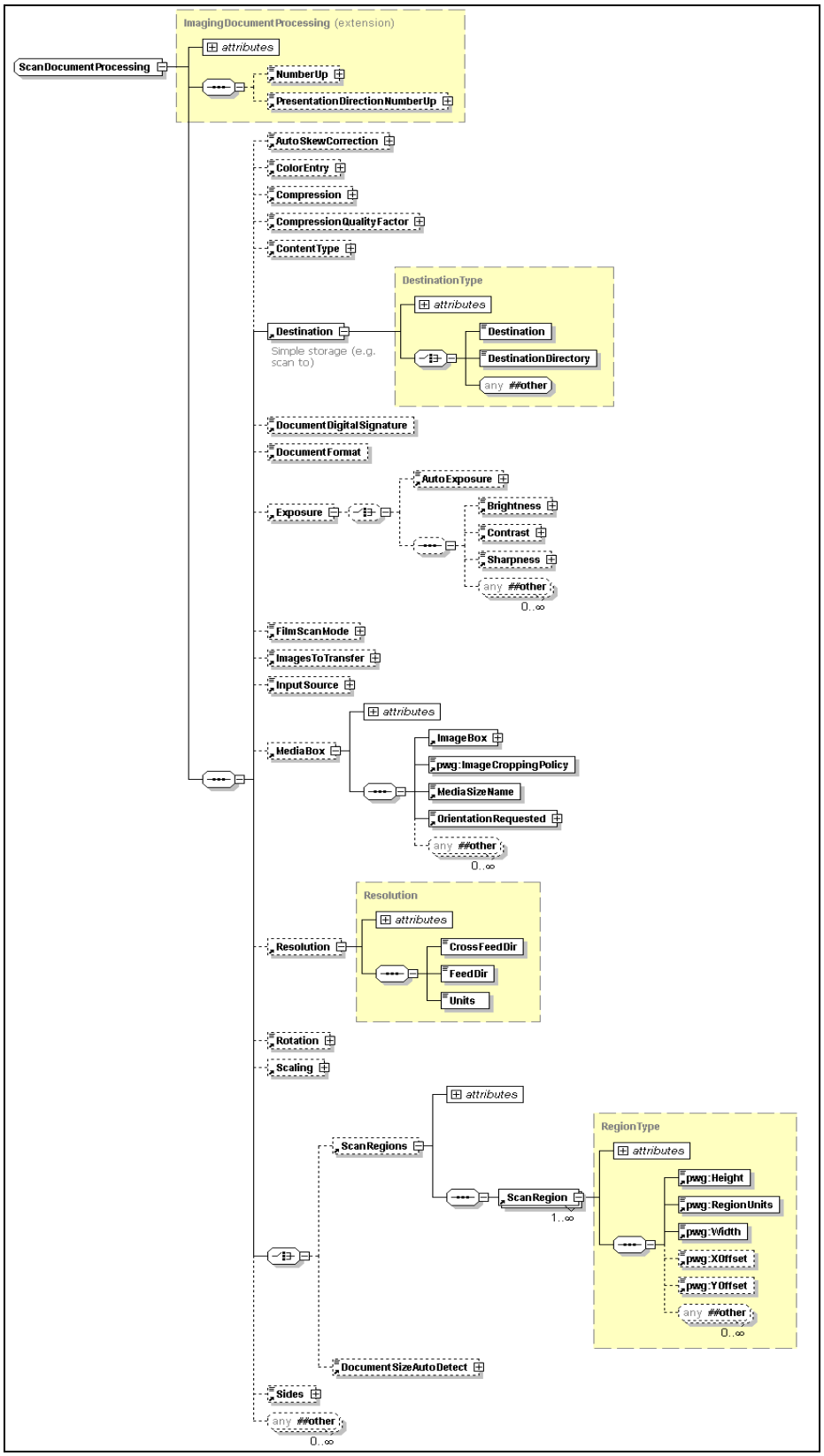


Figure 36 ScanDocumentProcessing

8.1.3.1.1 NumberUp

(int) This element indicates the number of Images from the input ScanRegions to be placed on to the finished output Image . *See* [rfc2911] §4.2.9

8.1.3.1.2 PresentationDirectionNumberUp

(keyword) This element is associated with the "NumberUp" element. This element specifies the placement order of the Images from the input ScanRegions on to the finished output Image. *(Keywords: TorightTobottom, TobottomToright, ToleftTobottom, TobottomToleft, TorightTotop, TotopToright, ToleftTotop) See* [PWG5100.3] §3.17

8.1.3.1.3 AutoSkewCorrection

(boolean) This element indicates if detection and correction of small skew orientation error from the media loading from the ADF or placement on the platen by the user is to be applied.

8.1.3.1.4 ColorEntry

(keyword) This element describes the color processing mode. Each keyword describes a color encoding, color space, bit depth and samples per pixel combination. Vendors may add additional keywords.

Keyword	Color Type	Color Encoding	Bit Depth	Samples per pixel
BlackandWhite1	Binary		1	1
Grayscale4	Gray		4	4
Grayscale8	Gray		8	8
Grayscale16	Gray		16	16
RGB24	color	RGB	24	8
RGB48	color	RGB	48	16
RGBa32	color	RGB	32	8
RGBa64	color	RGB	64	16
CMYK32	color	CMYK	32	8
CMYK64	color	CMYK	64	16

Table 4 – Color Processing Elements

8.1.3.1.5 Compression

(keyword) This element indicates the compression algorithm used on the Document Data, if any. *(Keywords: None, Deflate, Gzip, Compress) See* [rfc2911] §4.4.32

8.1.3.1.6 CompressionQualityFactor

(int) This element contains a normalized integer value used by JPEG compression to determine the amount of acceptable image loss. JPEG compression can be lossy, some amount of data is lost (not reproducible) or lossless. The higher the requested compression factor the smaller the resulting file size. The value is normalized as an integer between 0 and 100.

8.1.3.1.7 ContentType

(keyword) This element describes the main characteristics of the original Document. It is used as a hint to the Scan Service on how it should handle the scanning of the Hardcopy Document. *(Keywords: Auto, Text, Photo, Magazine, Halftone, Mixed)*

8.1.3.1.8 Destination

(complex) This element indicates where the output of the job SHALL be delivered. The destination is either a directory in which the Digital Document is stored or the URI to the Digital Document file.

8.1.3.1.8.1 Destination

(URI) This element is used when the output of the job is a Digital Document file.

8.1.3.1.8.2 DestinationDirectory

(URI) This element is used when the output of the job is a set of Digital Document files stored in adirectory.

8.1.3.1.9 DocumentDigitalSignature

(keyword) This element indicates the type of digital signature, if any, used in the creation of the Digital Document. *(Keywords: dss, none, pgp, smime, xmldsig)* See [PWG5100.7] §3.2.3 [PWG5100.5] §9.1.11

8.1.3.1.10 Document Format

(keyword) This element indicates the format used to save Scan Documents. This SHALL be an enumeration of MIME type which is pwg:DocumentFormat type. (*sample values: application/octet-stream, application/pdf, application/postscript, application/vnd.pwg-xhtml-print+xml, application/vnd.hp-PCL, image/g3fax, image/jpeg, image/tiff, image/tiff-fx, text/html, text/plain, text/plain; charset=ISO-8859-1, text/plain; charset=US-ASCII, text/plain; charset=utf-8, unknown*) See [rfc2911] §3.2.1.1 and [PWG5100.5] §9.1.12

8.1.3.1.11 Exposure

(complex) This element is a choice of the AutoExposure element or a sequence of four elements: Brightness, Contrast, Sharpness, and Any.

This specifies whether the Scan Service will automatically employ image processing techniques to reduce the background of the Document to a white Image. When set to *true* automatic background reduction will be performed on the input Document

8.1.3.1.11.1 AutoExposure

(boolean) When selected automatic background reduction will be performed on the input Document.

8.1.3.1.11.2 Brightness

(int) This element specifies the relative amount to enhance or reduce the brightness of the scanned Image. Negative values will darken the Image and positive values will lighten the Image. The value is normalized as an integer between -100 and 100.

8.1.3.1.11.3 Contrast

(int) This element specifies the relative amount to enhance or reduce the contrast of the scanned Image Negative values will reduce the apparent difference between light and dark pixels in the Image. Positive values will increase the apparent difference between light and dark pixels in the Image. The value is normalized as an integer between -100 and 100.

8.1.3.1.11.4 Sharpness

(int) This element specifies the relative amount to enhance or reduce object edges within a scanned Image. Positive values enhance object edges and negative values reduce object edges. The value is normalized as an integer between -100 and 100.

8.1.3.1.11.5 Any

This element provides an extension point for vendor differentiation and implementation specific extensions while maintaining interoperability

8.1.3.1.12 FilmScanMode

(keyword) This element specifies the exposure type of the film to be scanned. This element is only valid if the InputSource element is set to a value of 'FilmReader'. *(Keywords: NotApplicable, ColorSlideFilm, ColorNegativeFilm, BlackandWhiteNegativeFilm)*

8.1.3.1.13 ImagesToTransfer

(int) This element specifies the number of Images to scan from the Hardcopy Document.

8.1.3.1.14 InputSource

(keyword) This element specifies the source of the scanned Document*. (Keywords: Platen, ADF, FilmReader)*

8.1.3.1.15 MediaBox

(complex) This element is only applicable when the document format used to store the Digital Document has a similar construct that controls the placement and bounding of the scanned Image on the document format specific page Image. Prior to applying this element all other processing elements are assumed to have been applied. Note that origin of the Image does not move when an Image is rotated for the purposes of MediaBox. In other words the origin of the Image before rotation is in the upper left hand corner. After any amount of rotation the origin for the Image remains in the upper left hand corner. This is also true for the orientation of the MediaBox.
The Image is placed within the intersection of the MediaBox and the ImageBox under the control of the ImageCroppingPolicy. (See below)

8.1.3.1.15.1 ImageBox

(complex) This element describes the region where the Image from the scan of the ScanRegion will be placed.

8.1.3.1.15.1.1 Height

(int) This element identifies the height of the image box in RegionUnits.

8.1.3.1.15.1.2 RegionUnits

(keyword) This element identifies the units used for specifying the region. *(Keywords: Other, Unknown, TenThousandthsOfInches, Micrometers, Pixels, Percent)*

8.1.3.1.15.1.3 Width

(int) This element identifies the width of the image box in RegionUnits.

8.1.3.1.15.1.4 XOffset

(int) This element identifies the offset, positive or negative, on the X axis from the origin of the MediaBox in RegionUnits.

8.1.3.1.15.1.5 YOffset

(int) This element identifies the offset, positive or negative, on the X axis from the origin of the MediaBox in RegionUnits

8.1.3.1.15.1.6 Any

This element provides an extension point for vendor differentiation and implementation specific extensions while maintaining interoperability

8.1.3.1.15.2 ImageCroppingPolicy

(keyword)This element controls how to handle mismatches in the size of the Image and the intersection of the ImageBox and MediaBox where the Image will be placed.
Values are:

- 'Crop' – Image is not modified, areas outside the intersection are cropped, area that are not filled by the Image are left blank;
- 'StretchIsomorphicMax' – The Image is stretched maintaining its aspect ration until the both the height and width are filled, the Image outside the intersection is cropped;
- 'StretchIsomorphicMin - The Image is stretched maintaining its aspect ration until Image fills the height and width, no part of the Image is cropped;
- 'StretchNonIsomorphic' – The Image is stretched in the X direction until the width is filled, The Image is stretched in the Y direction until the height is filled, no part of the Image is cropped, The aspect ratio is not necessarily maintained

8.1.3.1.15.3 MediaSizeName

(keyword) This element identifies the size of the canvas for an Image in the Digital Document expressed as a media size name. . . *See [pwg5101.1] §5 for keyword values.*

8.1.3.1.15.4 OrientationRequested

(keyword) This element identifies the orientation of the MediaBox. *(Keywords: Landscape, Portrait, ReverseLandscape, ReversePortrait)* See [rfc2911] §4.2.10

8.1.3.1.16 Resolution

(complex) This specifies the resolution in the Feed and CrossFeed directions at which to capture the Image. The Resolution data structure does not limit the values. However, individual devices may limit the range of options to a predefined list or range. This information will be available as part of the ScanServiceCapabilities provided as a result of a GetScannerElements query. Resolution is specified in pixels per inch or pixels per centimeter. See printer-resolution in [RFC2911] §4.2.12

8.1.3.1.16.1 CrossFeedDir

(int) This specifies the resolution in Units for the CrossFeed directions at which to capture the Image See [rfc2911] §4.2.12

8.1.3.1.16.2 FeedDir

(int) This specifies the resolution in Units for the Feed directions at which to capture the Image See [rfc2911] §4.2.12

8.1.3.1.16.3 Units

(keyword) This specifies the units used to express the resolution *(Keywords: Dpcm, Dpi)* See [rfc2911] §4.2.12

8.1.3.1.17 Rotation

(keyword) Specifies the amount to rotate each Image of a scanned document specified in degrees clockwise *(Keywords: 0, 90, 180, 270)*

8.1.3.1.18 Scaling

(complex) This element specifies the scaling to be applied to the scanned Image. Isomorphic (i.e. the aspect ratio of the Image does not change) is accomplished by specifying the same values for ScalingWidth and ScalingHeight. *A value of '100' specifies that no adjustments are made to the scanned Image. Magnification is expressed in 1 percent increments. Values below '100' reduce the magnification and values above increase magnification.*

8.1.3.1.18.1 ScalingWidth

(int) This element specifies the scaling in the fast scan direction *expressed in 1 percent increments. Values below '100' reduce the magnification and values above increase magnification.*

8.1.3.1.18.2 ScalingHeight

(int) This element specifies the scaling in the slow scan direction *expressed in 1 percent increments. Values below '100' reduce the magnification and values above increase magnification.*

8.1.3.1.18.3 AutoScaling

(boolean) This element specifies the automatic adjustment of the Scaling of the image from the Hardcopy Document in an implementation specific manner.

8.1.3.1.19 Scan Region

(complex)This element specifies the area within the input Hardcopy Document's media sheet side boundaries to scan.
If ScanRegion is not specified, the device should use 0 as the offsets and the width and height of the InputSize, if given. If ScanRegion is not specified and InputSize is not specified or cannot be determined by the device, the implementation is up to the hardware vendor. This element is mutually exclusive with DocumentSizeAutoDetect.

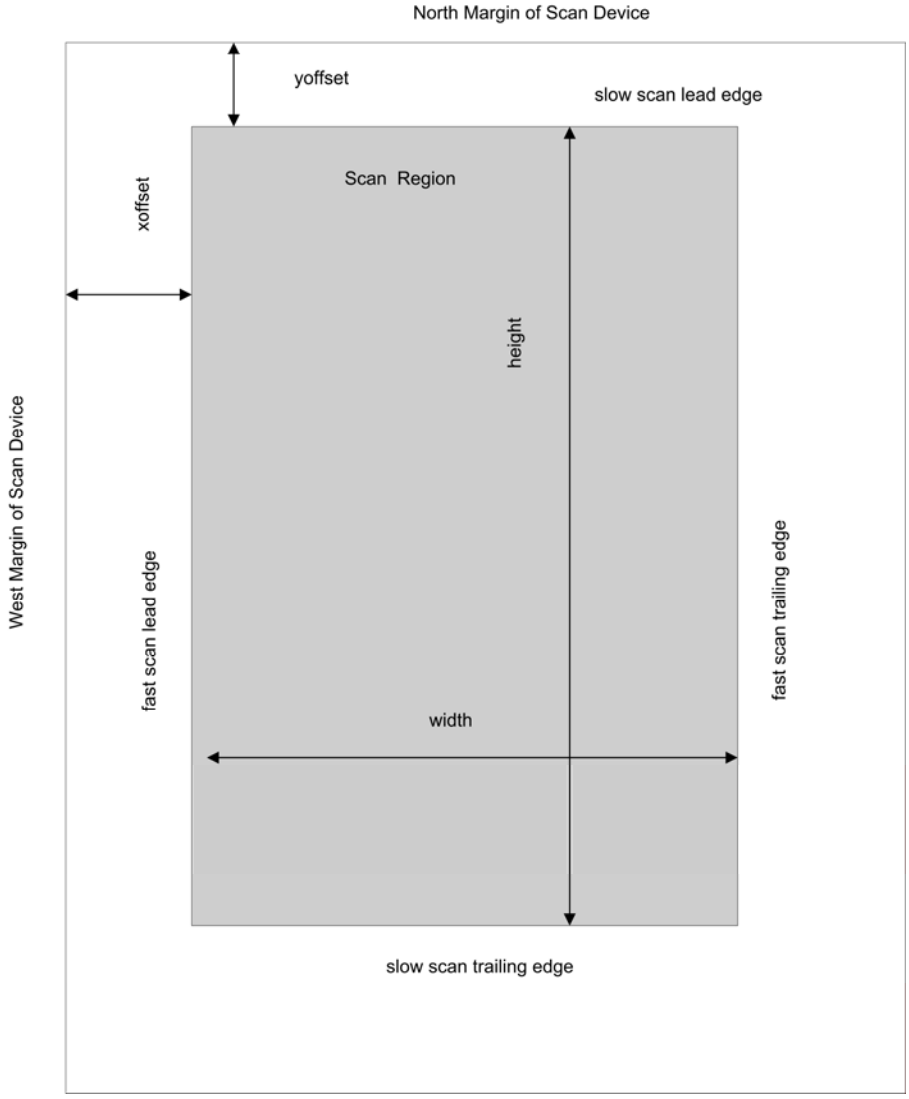


Figure 37 ScanRegion

Width corresponds to the fast scan direction and Height corresponds to the slow scan direction.

8.1.3.1.19.1.1 Height

(int) This element identifies the height of the ScanRegion in RegionUnits

8.1.3.1.19.1.2 RegionUnits

(keyword) This element identifies the units used for specifying the region. *(Keywords: Other, Unknown, TenThousandthsOfInches, Micrometers, Pixels, Percent)*

8.1.3.1.19.1.3 Width

(int) This element identifies the width of the ScanRegion in RegionUnits

8.1.3.1.19.1.4 XOffset

(int) This element identifies the offset, positive or negative, on the X axis from the West Margin of the Scanner Subunit in RegionUnits

8.1.3.1.19.1.5 YOffset

(int) The offset, positive or negative, on the X axis from the West Margin of the Scanner Subunit in RegionUnits

8.1.3.1.20 DocumentSizeAutoDetect

(keyword) This element specifies that the area within the input Hardcopy Document's media sheet side boundaries to scanare automatically determined by the device in an implementation specific manner. This element is mutually exclusive with ScanRegion.

8.1.3.1.21 Sides

(keyword) This element indicates how an Image is to be scanned from the side(s) of the Hardcopy Document. *(Keywords: OneSided, TwoSidedLongEdge, TwoSidedShortEdge, TwoSidedLongEdge) See* [rfc2911] §4.2.8

8.1.3.1.22 Any

This element provides an extension point for vendor differentiation and implementation specific extensions while maintaining interoperability

8.1.3.2 Scan Job Description

(complex – ScanJobDescription) Figure 38 is a view of the Description elements for the Scan Job. Note that ScanJobDescription consists of two sequences. The first represents elements inherited from the Imaging Job super class (i.e. ImagingJobDescription) and includes elements such as JobName. The second sequence includes Scan Service specific extensions to the super class. Currently the only extension is the extension point for Vendors. These elements are set by the Scan Client during job creation.

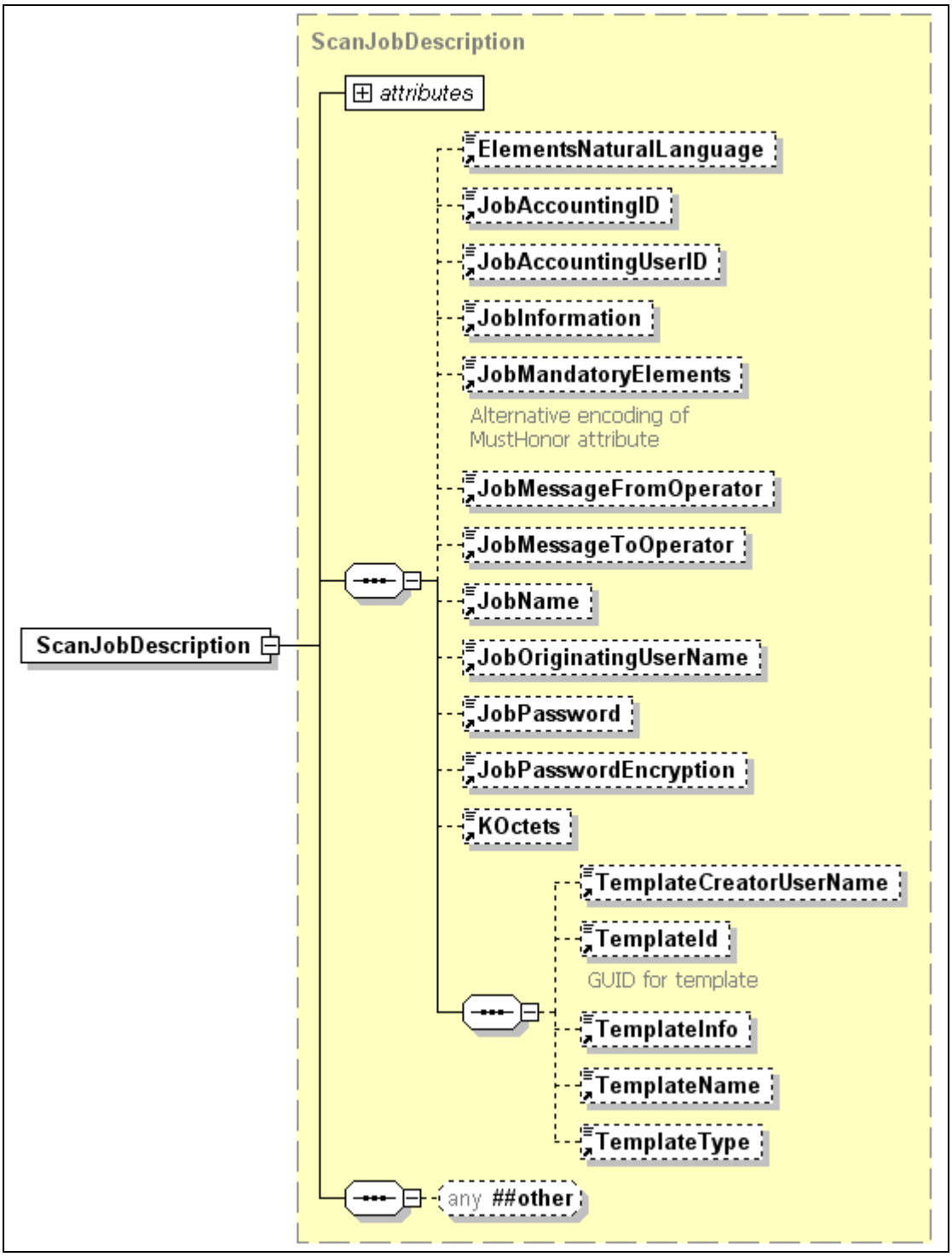


Figure 38 ScanJobDescription

8.1.3.2.1 ElementsNaturalLanguage

(keyword) This element indicates the natural language of the elements with string syntax that were set by the End User. ElementsNaturalLanguage is the language for system generated character strings. For example, the response of a request to Scan Service can be in the language you specified here(example values: en-us, fr, de, ja) See [rfc3066].

8.1.3.2.2 JobAccountingID

(string) This element specifies the accounting ID associated with this Job. See [PWG5100.3] §3.6

8.1.3.2.3 JobAccountingUserID

This element specifies the User ID associated with the "JobAccountId". See [PWG5100.3] §3.7

8.1.3.2.4 JobMandatoryElements

(list of keyword) This element allows a user to list which Ticket elements the Scanner must honor. The Scan Service rejects the request for job creation if any of the listed elements are unsupported or contain values that the Scan Service does not support. All of the remaining supplied elements are best effort. See attribute-fidelity [rfc2911] §15.1 and [PWG5100.5] §8.1.1

8.1.3.2.5 JobMessageFromOperator

(string) This element is a message to the End User indicating the reasons for any management action taken on this Job See [rfc2911] §4.3.16

8.1.3.2.6 JobMessageToOperator

(string) This element is a message from the End User to indicate something about the processing of this Job. See [PWG5100.3] §3.10

8.1.3.2.7 JobName

(string) The Scan Service sets this to the client-supplied end-user friendly name for the Job. When it is not supplied by the client, the Scan Service must generate a name from other information. [rfc2911] §4.3.5

8.1.3.2.8 JobOriginatingUserName

(string) The Scan Service sets this element to the most authenticated printable name that it can obtain (example: "John Doe", \authDomain\John Doe") See [rfc2911] §4.3.6

8.1.3.2.9 JobPassword

(octetString) The element contains a password supplied by the client encrypted according to method specified by the client in the JobPasswordEncryption element. See [prod-print2] §4.1

8.1.3.2.10 JobPasswordEncryption

(keyword) The element specifies the type of encryption that the client is used for the supplied value of the JobPassword element. (Keywords: None, TripleDES, AES, ECC, Md2, Md4, Md5, Sha) See [prod-print2] §4.2

8.1.3.2.11 KOctets

(int) This element is the total size of this Job's Document(s) in integral units of 1024 octets. See [rfc2911] §4.3.17.1

8.1.3.2.12 TemplateCreatorUserName

(string) This element is the name of user who created the template used to create the ScanJobTicket (if any). Note: For a Job Template this is used to hold the most authenticated user name of the Template Creator.

8.1.3.2.13 TemplateId

(URI) This element is the ID of the template used to create the ScanJobTicket (if any).

8.1.3.2.14 TemplateInfo

(string) This element is the TemplateInfo of the template used to create the ScanJobTicket (if any). .

8.1.3.2.15 TemplateName

(string) This element is the name of template used to create the ScanJobTicket (if any). Note: For a Job Template this is used to hold the Template Repository wide unique Template name.

8.1.3.2.16 TemplateType

(keyword) This element is the type of the template used to create the ScanJobTicket (if any). The type of the template MUST be appropriate for the service which in this case would be 'ScanJob' or ScanDocument'.

8.1.3.2.17 Any

This element provides an extension point for vendor differentiation and implementation specific extensions while maintaining interoperability

8.1.3.3 Scan Job Processing

(complex – ScanJobProcessing) The ScanJobProcessing 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 Scan Service does not process the job unless the element is supported, otherwise the Scan Service processes the job with its best effort.

Note that ScanJobProcessing consists of two sequences. The first represents elements inherited from the Imaging Service super class and includes elements such as JobHoldUntil. The second sequence includes Scan Service specific extensions to the super class such as BatchModeScan.

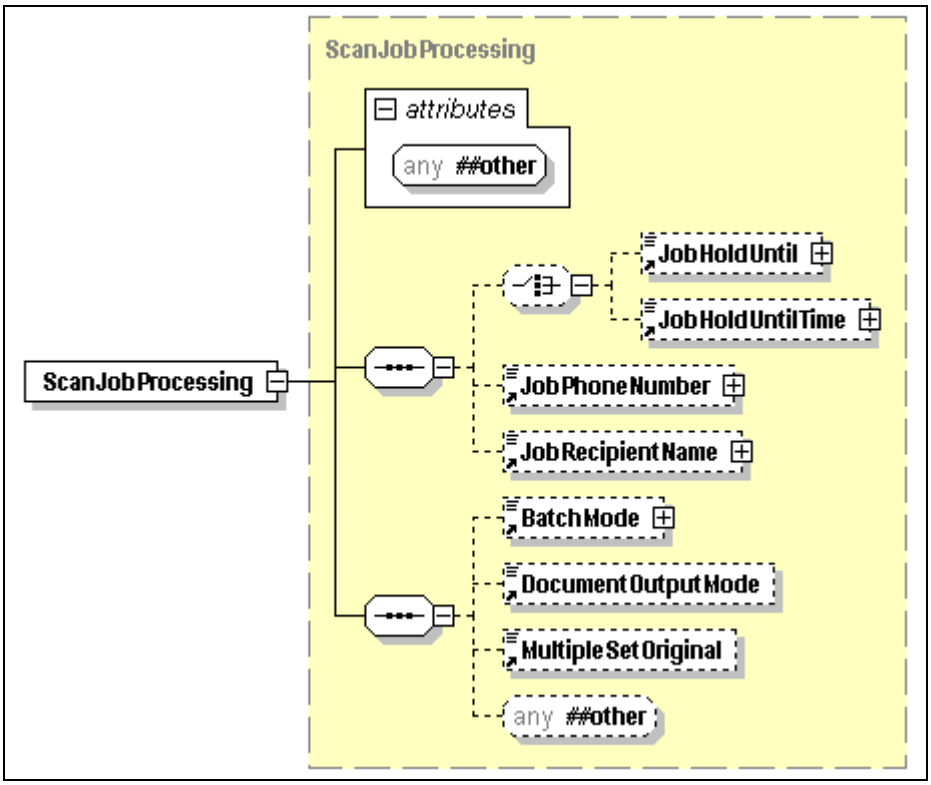


Figure 39 ScanJobDescription

8.1.3.3.1 JobHoldUntil

(keyword) This element is the duration of time a job is put on hold, i.e. prohibited from being scheduled for processing. Note that if this element is specified JobHoldUntilTime can not be specified. See [rfc2911] §4.2.2

8.1.3.3.2 JobHoldUntilTime

(DateTime) This element is the absolute date and time a job will be released after being put on hold. This element allows you to hold a remotely submitted job until a specific time for processing. Note that if this element is specified JobHoldUntil can not be specified. See [prod-print2] §5.4

8.1.3.3.3 JobPhoneNumber

(URI) This element contains the contact phone number for the owner of the Job. This informational element's value can be used to contact the owner in the event additional information is required (e.g. what to do if the quality of the scan is deemed unacceptable) See [prod-print2] §5.5, [RFC3966], [RFC[4355]

8.1.3.3.4 JobRecipientName

(string) This element contains the name of the recipient for the job. See [prod-print2] §5.6

8.1.3.3.5 BatchMode

(boolean) This element's value indicates whether the job is to be processed in Batch Scan Mode. Batch mode scanning consists of a set of Hard Copy Documents that are placed in a document feeder. The set of Hard Copy Documents will contain Physical ScanJobTicket(s) and Physical ScanDocumentTicket(s) that will control the processing of the stack of Hard Copy Documents.

8.1.3.3.6 DocumentOutputMode

(keyword) This element controls the number of Documents and output files generated from a Scan Job. (Keywords: SingleDocumentSingleFile, SingleDocumentMultipleFile, MultipleDocumentSingleFile, MultipleDocumentMultipleFile) See §6.2

8.1.3.3.7 MutipleSetOriginal

(boolean) This element controls whether the Scan Service is required to intake multiple sets of originals from the selected input source.

8.1.3.3.8 Any

This element provides an extension point for vendor differentiation and implementation specific extensions while maintaining interoperability

9 Scan Document Model definition

(ScanDocuments is a list of complex –ScanDocument) This section provides the detailed descriptions of the elements that comprise the Scan Document model.

Figure 40 is the top level view of the Scan Document. Scan Documents are associated with one Job. The state of the Document is described in the ScanDocumentStatus element. ScanJobTicket 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 ScanDocumentTicket with the Scan Document. The ScanJobTicket and ScanDocumentTicket represent the End User's intent while the Job Receipt and Document Receipt represent what the Scan Service actually did.

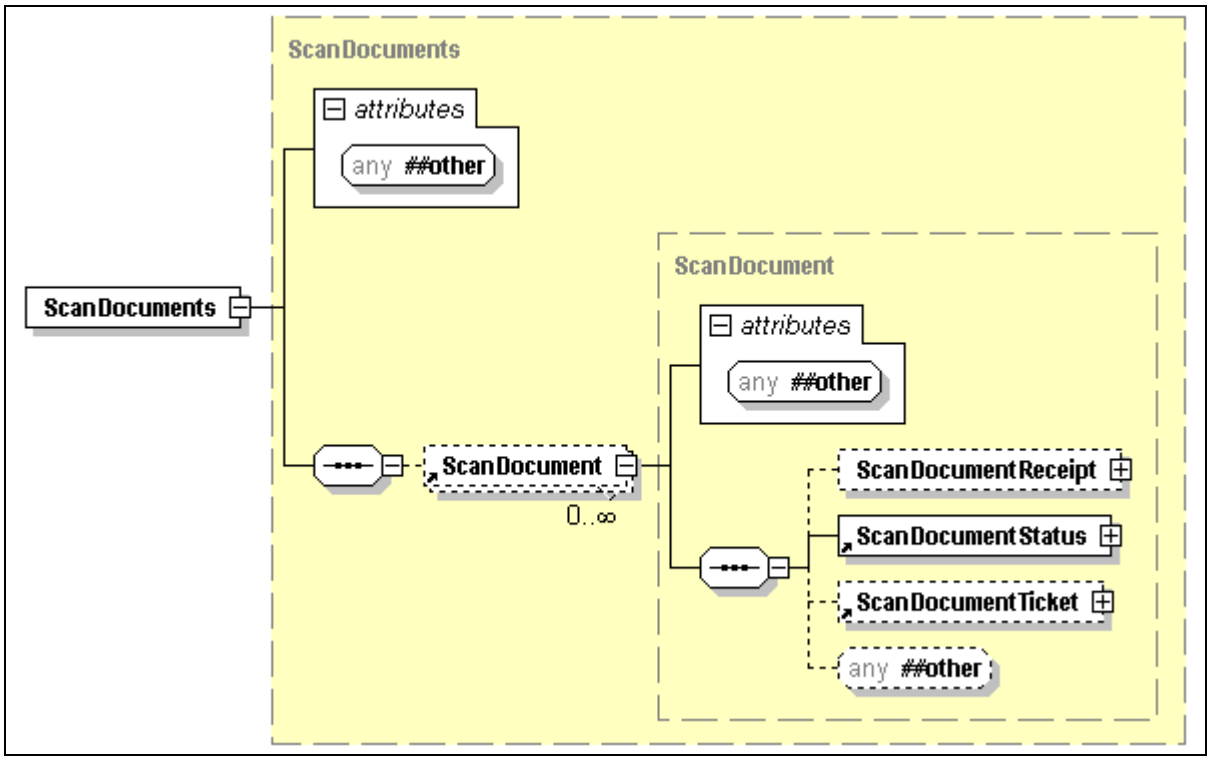


Figure 40 High Level Scan Document View

9.1.1 Scan Document Receipt

(complex – ScanDocumentReceipt) This element has exactly the same structure as the ScanDocumentTicket. For each processing element of a Scan Document, it records the actual value used by the Scan Service for processing the Scan Document. It contains the elements supplied by the Scan Client, Any substitutions made by the Scan Service and any Default elements applied by the Scan Service. See §9.1.3 for element descriptions.

9.1.2 Scan Document Status

(complex – ScanDocumentStatus) Figure 41 is a view of the Status elements for the Scan Document. Note that ScanDocumentStatus consists of two sequences. The first represents elements inherited from the Imaging Document super class (i.e. ImagingDocumentStatus) and includes elements such as DocumentNumber and DocumentState. The second sequence includes Scan Service specific extensions to the super class such as the Scan Document counters.

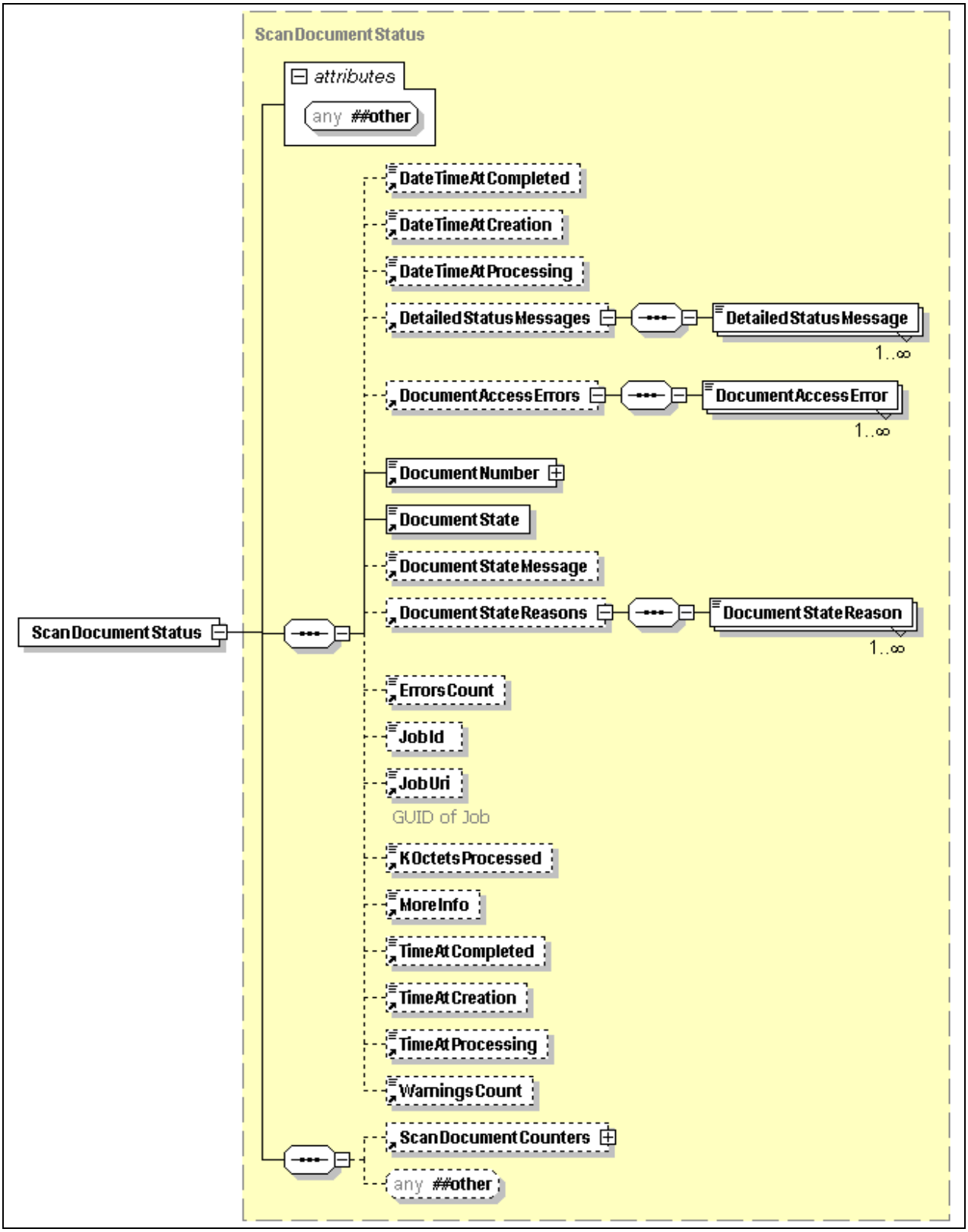


Figure 41 Scan Document Status

9.1.2.1 DateTimeAtCompleted

(DateTime) This element indicates the date and time at which the Job object completed (or was canceled or aborted). See [rfc2911] §4.3.14.7

9.1.2.2 DateTimeAtCreation

(DateTime) This element indicates the date and time at which the Job object completed created. See [rfc2911] §4.3.14.5

9.1.2.3 DateTimeAtProcessing

(DateTime) This element indicates the date and time at which the Job object first began processing (i.e. scanned). See [rfc2911] §4.3.14.6

9.1.2.4 DetailedStatusMessages

(list of string) This element specifies additional detailed and technical information about the job. The Scan Service MAY localize the message(s), since they are intended for use by the system administrator or other experienced technical persons. Localization might obscure the technical meaning of such messages. Clients MUST NOT attempt to parse the value of this attribute. See [rfc2911] §4.3.10

9.1.2.5 DocumentAccessErrors

(list of string) This element provides additional information about each document access error for this job encountered by the Scan Service after it returned a response to scan initiation operation and subsequently attempted to store the Image data to the locations supplied in the ScanJobTicket. For errors in the protocol associated with a URI scheme, such as 'http:' or 'ftp:', the error code is returned in parentheses, followed by the URI. For example:
(404) http://ftp.pwg.org/pub/pwg/mfd/job1207.png See [rfc2911] §4.3.11

9.1.2.6 DocumentNumber

(int) This element uniquely identifies a Document within a Job. The first Document is number ‘1’. Subsequent Documents are assigned a monotonically increasing DocumentNumber. Once assigned a Document’s DocumentNumber is invariant. See [PWG5100.4] §9.2, [PWG5100.5] §9.1.23

9.1.2.7 DocumentState

(keyword) This element identifies the current state of Scan Document. The state values MUST NOT be extended by an implementation. See [PWG5100.5] §9.1.25
Values:

From rfc2911:

- **Aborted** – The Document has been aborted by the system, usually while the Document was in the 'Processing' state and the Scan Service has completed aborting the Document and all Document status elements have reached their final values for the Document. While the Scan Service is aborting the Document, the Document remains in its current state, but the Document’s " DocumentStateReasons” element SHOULD contain the 'ProcessingToStopPoint' and 'AbortedBySystem' values. When the Document moves to the 'Aborted' state, the 'ProcessingToStopPoint' value, if present, MUST be removed, but the 'AbortedBySystem' value, if present, MUST remain

- **Canceled** – The Document has been canceled as a result of the Scan Job being cancelled by a CancelScanJob” operation prior to the Document reaching a ‘Completed’ or ‘Aborted’ state. While the Scan Service is canceling the Document, the Document remains in its current state, but the Document’s "DocumentStateReasons” element SHOULD contain the 'ProcessingToStopPoint' value and one of the 'CanceledByUser', 'CanceledByOperator', or 'CanceledAtDevice' values. When the Document moves to the 'canceled' state, the 'ProcessingToStopPoint' value, if present, MUST be removed, but the 'CanceledByXxx', if present, MUST remain

- **Completed** – The Document has completed successfully or with warnings or errors after processing and all of the Document’s Digital Documents have been sent to their Destination and all Document status elements have reached their final values for the Document. The Document’s "

DocumentStateReasons" element SHOULD contain one of: 'CompletedSuccessfully', 'CompletedWithWarnings', or 'CompletedWithErrors' values.

- **Pending** – The Document has not started to be processed at all
- **Processing** – The job has begun using, or is attempting to use, one or more purely software processes that are analyzing, creating, or interpreting the Hard Copy Document or Digital Document. The job has begun using, or is attempting to use, one or more hardware devices that are analyzing, creating, or interpreting the Hard Copy Document or Digital Document. The Digital Document is ready for sending to its Destination, but the Output Channel is not yet transferring it, either because the Digital Document hasn't reached the Output Channel or because the Document is queued in the Output Channel or some other spooler, awaiting the Output Channel to transfer it. The 'processing' state for the Document indicates that the Document has begun to be processed. Even if the Job stops being processed, the Document remains in the 'processing' state until it moves to one of the three terminal states. Implementations MAY include additional values in the Document's "DocumentStateReasons" element to indicate the progress of the Document, such as adding the 'Scanning' value to indicate when the Scanner device is actually acquiring the Image from the ScanRegion of the Hard Copy Document .

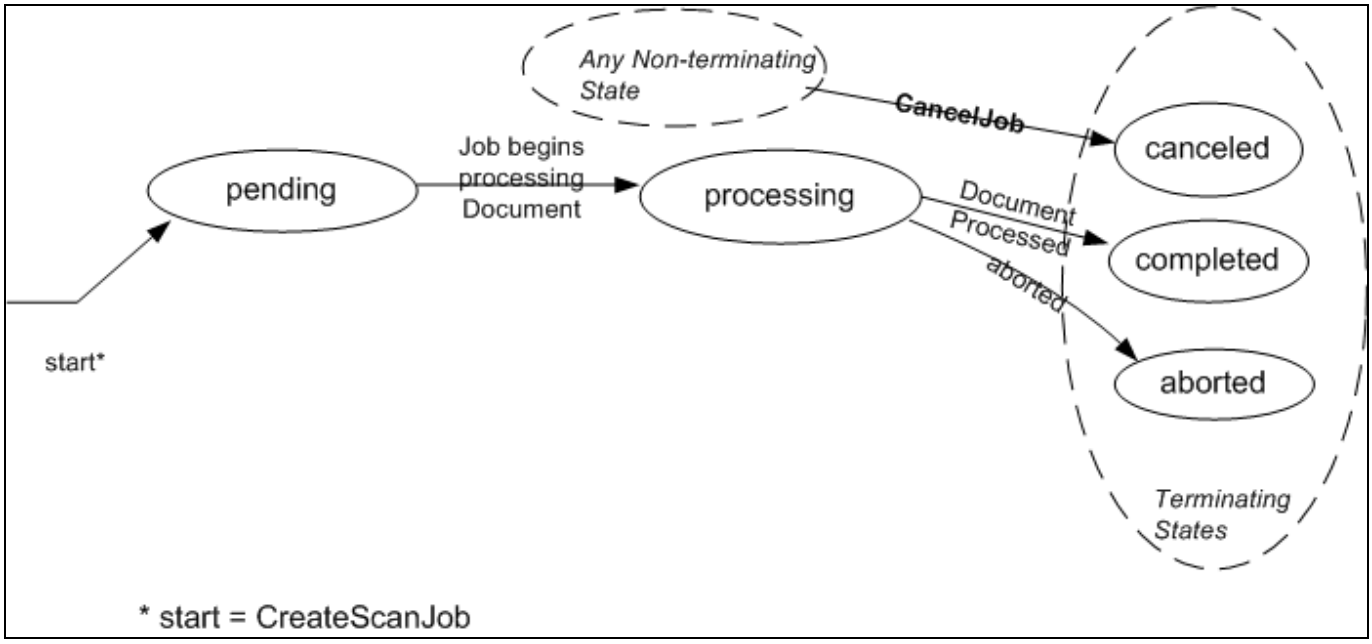


Figure 42 Document State Transition Diagram

9.1.2.8 DocumentStateMessage

(string) This element provides information about the Document State and StateReasons in human readable text. If the Service supports this element, it MUST be able to generate the messages in any of the natural languages supported by the Service. See [PWG5100.5] §9.1.26

9.1.2.9 DocumentStateReasons

(list of keyword) This element supplies additional detail about the Document state. The scan specific value of 'Scanning" has been added See [PWG5100.5] §9.1.27 and {RFC2911] §4.3.8 for standard values.

9.1.2.10 ErrorsCount

(int) This element is the number of errors encountered while processing the Document See [PWG5100.5] §9.1.29

9.1.2.11 JobId

(int) JobId is an integer value that identifies the job to which this Document belongs. See [PWG5100.5] §9.1.18

9.1.2.12 JobUri

(URI) This element is the globally unique ID of the job to which this Document belongs See [PWG5100.5] §9.1.19

9.1.2.13 KOctetsProcessed

(int) This element is the total number of octets processed in integral units of 1024 octets so far See [rfc2911] §4.3.17.1

9.1.2.14 MoreInfo

(string) This element is the URI used to obtain information intended for End User consumption about this specific Job See [rfc2911] §4.3.4

9.1.2.15 TimeAtCompleted

This element is the time at which the Document completed in “PrinterUpTime” seconds See [rfc2911] §4.3.14.3

9.1.2.16 TimeAtCreation

(int) This element is the time at which the Document was created in “PrinterUpTime” seconds See [rfc2911] §4.3.14.1

9.1.2.17 TimeAtProcessing

(int) This element is the time at which the Document first began processing in “PrinterUpTime” seconds See [rfc2911] §4.3.14.2

9.1.2.18 WarningsCount

(int) This element is the total number of warnings that a Scan Service has generated while processing and storing the Job’s Document(s). See [PWG5100.4 §6.1

9.1.2.19 ScanServiceCounters

(complex) This element contains the counters for the amount of work performed for this Document by the Scan Service, timers covering utilization and monitoring information covering errors, warnings , traffic, job counts are included. See [PWG5106.1-2007].

9.1.2.20 Any

This element provides an extension point for vendor differentiation and implementation specific extensions while maintaining interoperability

9.1.3 ScanDocumentTicket

(complex – ScanDocumentTicket) The ScanDocumentTicket contains description and processing elements provided by the Scan client during Scan Document creation. This information is used by the Scan Service during the processing of a Scan Document. This information is made available to Scan Clients through the GetScanDocumentElements operation.

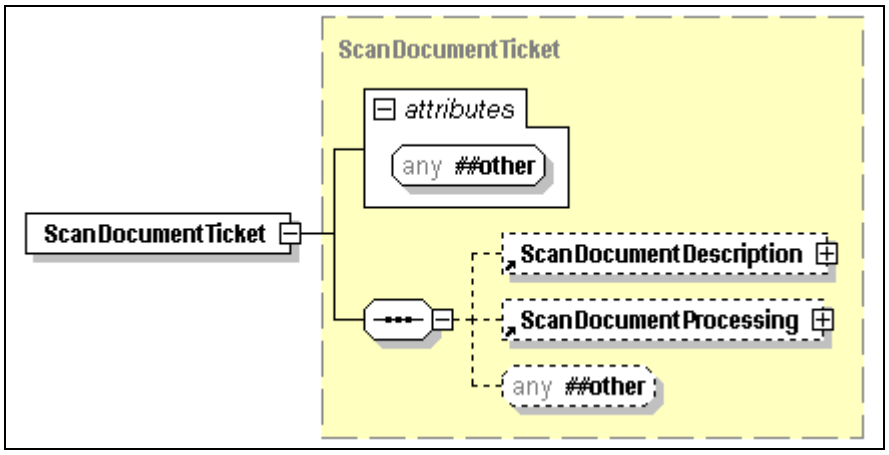


Figure 43 ScanDocumentTicket

9.1.3.1 Scan Document Description

(complex – ScanDocumentDescription) Figure 44 is a view of the Description elements for the Scan Document. Note that ScanDocumentDescription consists of two sequences. The first represents elements inherited from the Imaging Document super class (i.e. ImagingDocumentDescription) and includes elements such as DocumentName. The second sequence includes Scan Service specific extensions to the super class. Currently the only extension is the extension point for Vendors. These elements are set by the Scan Client during job creation.

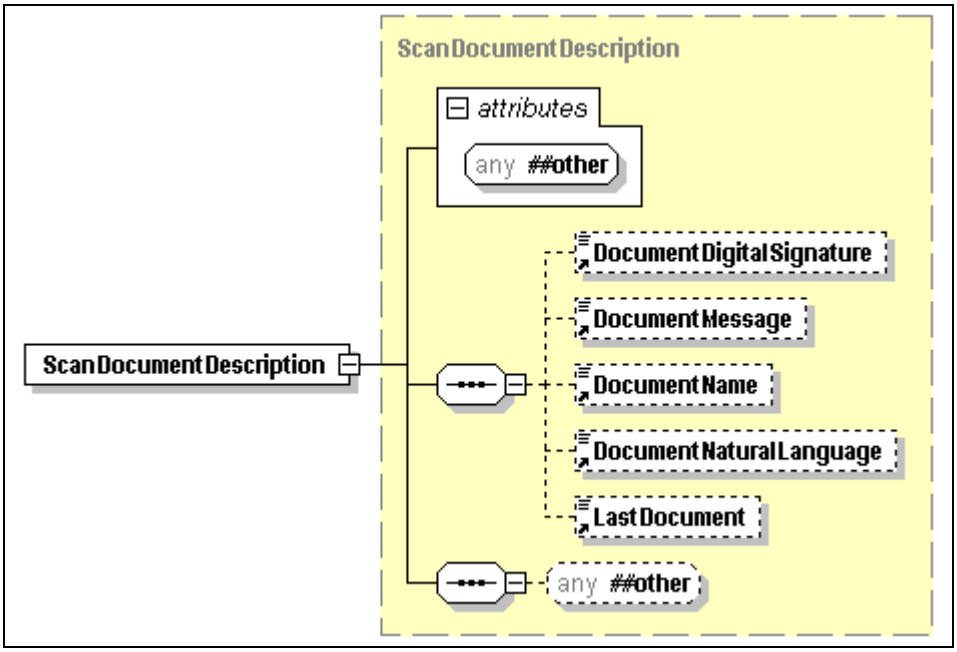


Figure 44 ScanDocumentDescription

9.1.3.1.1 DocumentDigitalSignature

(keyword) This element indicates the type of digital signature that is used in the creation of the Digital Document. *(Keywords:* dss, none, pgp, smime, xmldsig*).* See [PWG5100.7] §3.2.3 [PWG5100.5] §9.1.11

9.1.3.1.2 DocumentMessage

(string) This element is a message from either (1) the user to the operator about the Document or (2) from the operator, system administrator, or "intelligent" process to indicate to the End User the reasons for modification or other management action taken on the Document. See [PWG5100.5] §9.1.20

9.1.3.1.3 DocumentName

(string) This element is the name for this Document to be used in an implementation specific manner. See [rfc2911] §3.2.1.1

9.1.3.1.4 DocumentNaturalLanguage

(keyword) This element provides a hint about the language used in Hard Copy Documents that contain text. This element is not useful for Hard Copy Documents that do not contain text or are not intended to undergo Optical Character Recognition (OCR). See [rfc2911] §3.2.1.1, [PWG5100.5] §9.1.22

9.1.3.1.5 LastDocument

(boolean) The value of this element is set to 'true' for the last Document in the job. Otherwise the value is set to 'false' or the element is omitted. See [rfc2911] §3.3.1

9.1.3.1.6 Any

This element provides an extension point for vendor differentiation and implementation specific extensions while maintaining interoperability

9.1.3.2 Scan Document Processing

This group element has exactly the same structure as the Scan Document Processing element of Scan Job (See 8.1.3.1). 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.

10 Scan Service Theory of Operation

The Scan Service operates autonomously through three phases: initialization, online, and offline.

At start-up the Scan Service enters its initialization phase that initializes all its service attributes and connected subunits. This phase may include tests of the associated Subunits and self-testing of the Scan Service itself. After the initialization and tests are successful, the Scan Service enters the online phase with a state of "Idle". The Scan Service is ready for service discovery and accepting service requests from Scan Clients. The Scan Service may authenticate and register itself with a service directory or announces its service to the network domain in which it resides.

The Scan Service accepts new requests as long as it's not disabled and is in one of the three states: Idle, Processing or Stopped. Performing an administrative Disable() operation while in any state will stop the Scan Service from accepting new jobs. Performing an Enable() operation in any state while the Scan Service is disabled will enable new jobs to be accepted again.

A user submits a Scan Job through a local (via MFD UI) or remote (via local network or Internet) Scan Client to a selected target Scan Service that has the desired scan capabilities. While the service is enabled, a Scan Client can request any Scan Service operations specified in Sections 11.1 and 11.1.8.1. A Scan Client uses the CreateScanJob operation to submit a Scan Job on behalf of a user. The Scan Service places all submitted jobs in the ActiveJobs queue and schedules jobs for processing immediately or when a StartJob event is signaled based on job priority. A user may specify a JobHoldUntilTime in the Scan Job's Ticket for a remotely submitted Scan Job to allow ample time for user to walk up to the scanner for placing his/her Hardcopy originals on the scanner. An administrator can also put a Scan Job in the ActiveJobs queue on hold via a HoldScanJob() operation preventing it being scheduled and a ReleaseJob() operation will release the Scan Job for scheduling again.

When a Scan Job is released for scheduling and reaches the top of ActiveJobs queue, the Scan Service enters or remains in its Processing state. During job processing, the Scan Service can be interrupted by a "PauseScanService()" operation to enter the "Stopped" state. This allows a user to submit and process an urgent Scan Job or a job for another service, and a Resume() operation resumes previous Scan Job processing afterwards. Upon completion of a Scan Job the Scan Service moves the Scan Job from the ActiveJobs queue to the JobHistory queue.

When there are critical conditions impacting Scan Serviceability during "Idle" or "Processing" state, either a E.Critical event is generated or an Administrative PauseScanService() is performed to bring the service to the Stopped state. From there the condition can be fixed by user's intervention. Then either the Scan Service generates a E.CriticalCleared event or an administrator performs a Resume() operation to bring the Scan Service back to "Idle" or "Processing" state. Otherwise, if the Scan Service needs a ShutdownScanService() operation followed by a restart or ShutdownScanService() for testing, both will require a StartupScanService() operation to bring the service back to "Idle" state and then job processing may continue.

The lifecycle for a Scan Job begins when it is created by the Scan Service on behalf of a user issuing a CreateScanJob request. The newly created Scan Job is placed on the ActiveJobs queue. The state of the Scan Job is either 'Pending' or, if the request contained a JobHoldUntilTime in the Scan Job's Ticket, 'PendingHeld'. When the conditions are met to release a 'PendingHeld' Scan Job, its state transitions to 'Pending'. Scan Jobs may be held and released through administrative operations. When a Scan Job reaches the top of the ActiveJobs queue it is scheduled and the state of the Scan Job transitions to 'Processing'. If for any reason the Scan Service becomes 'Stopped' the state of a processing Scan Job becomes 'ProcessingStopped'. When the Scan Service state returns to 'Processing' the Scan Job state returns to 'Processing'. Upon completion the status of the Scan Job becomed 'Completed'. It is also possible for a Scan Job to fail. This causes the Scan Job state to transition to 'Aborted'. At any time all Scan Jobs in the ActiveJobs queue, whether being held, pending for scheduling, in processing, or being temporarily stopped from processing, can be canceled via a CancelScanJob() operation by an authorized user. The Scan Job state will then transition to 'Canceled' Any Scan Job reaching a terminating state of 'Completed', Canceled' or 'Aborted' is moved from the ActiveJobs queue to the JobHistory queue.

11 Scan Service Interfaces

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

11.1 Basic Scan Service Operations

11.1.1 CancelScanJob

This REQUIRED operation allows a client to cancel a Scan Job from the time the job is created up to the time it is completed, canceled, or aborted. Since a Job might already be scanning by the time a CancelScanJob is received, some Images might be stored to the Destination before the job is actually terminated.

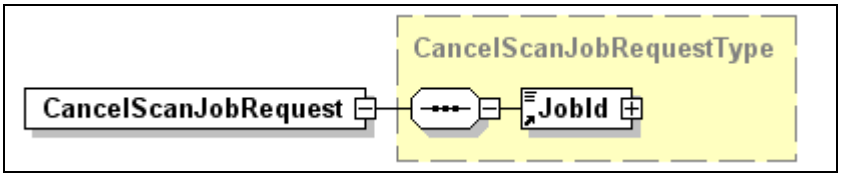
The Scan Service MUST accept or reject the request based on the job's current state and transition the job to the indicated new state as follows:

Current JobState	New JobState	status	Note
Pending	Canceled	Success	
PendingHeld	Canceled	Success	
Processing	Canceled	Success	
Processing	Processing	Success	See Note 1
Processing	Processing	Fault	See Rule 2
Processing Stopped	Canceled	Success	
Processing	Processing	Success	See Note 1

Stopped	Stopped		
Processing Stopped	Processing Stopped	Fault	See Rule 2
Completed	Completed	Fault	
Canceled	Canceled	Fault	
Aborted	Aborted	Fault	

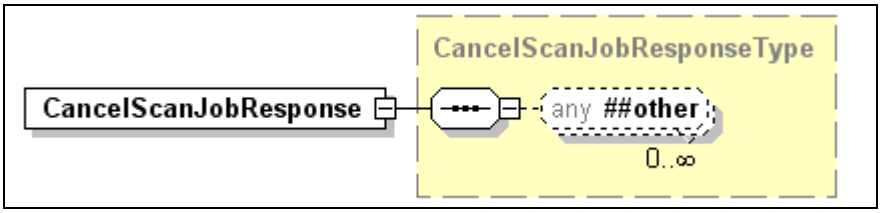
Note 1: If the implementation requires some measurable time to cancel the job in the ‘Processing’ or ‘ProcessingStopped’ states, the Scan Service MUST add the ProcessingToStopPoint ' value to the job's "JobStateReasons" element and then transition the job to the 'Canceled' state when the processing ceases.
Note 2: If the Job already has the 'ProcessingToStopPoint' value in its "JobStateReasons" element, then the Scan Service MUST reject a CancelScanJob operation.

11.1.1.1 CancelScanJobRequest



11.1.1.2 CancelScanJobResponse

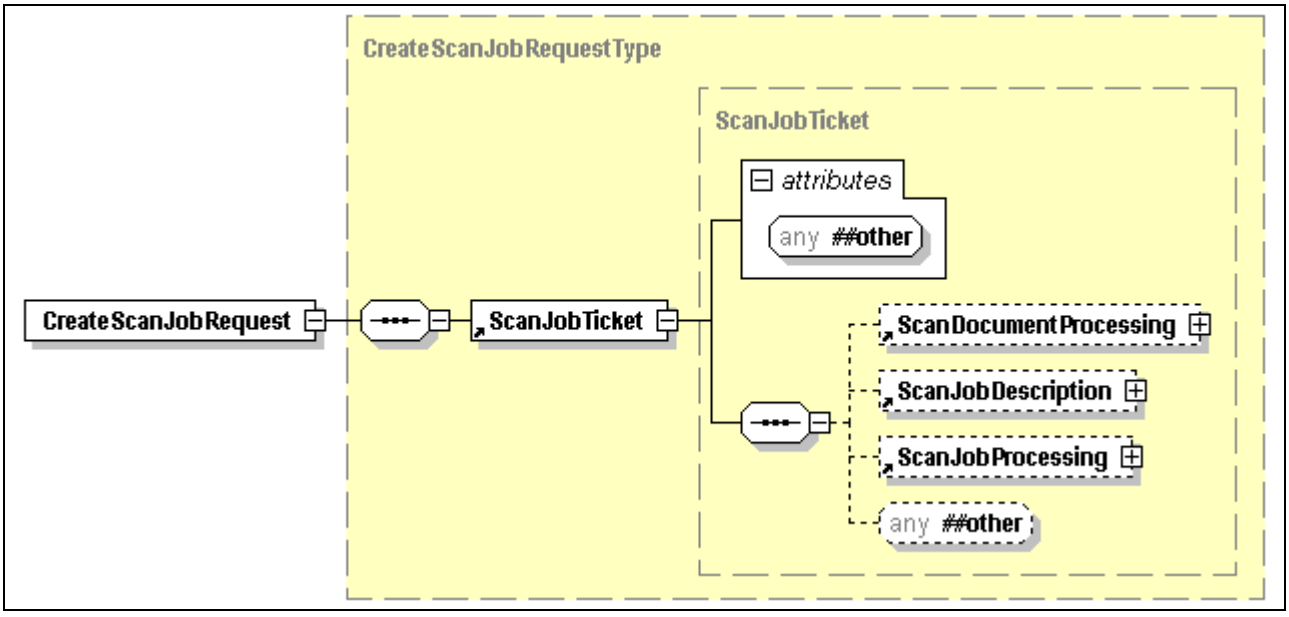
Note: The response is expected to be an empty successful return or a fault. The response below indicates that vendors may extend the response with additional information



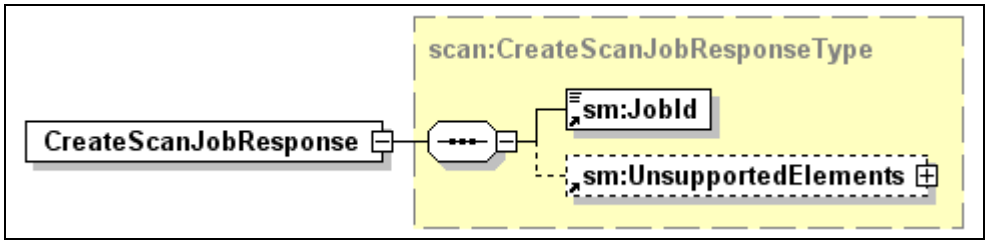
11.1.2 CreateScanJob

This REQUIRED operation allows a Scan Client to create a Scan Job on the Scan Service. Upon completion the Scan Job is available for scheduling unless a Job Processing instruction (e.g. JobHoldUntil) explicitly prevents it. This operation MUST fail the Service’s “isAcceptingJobs” element value is ‘false’. (See 11.2.1)

11.1.2.1 CreateScanJobRequest



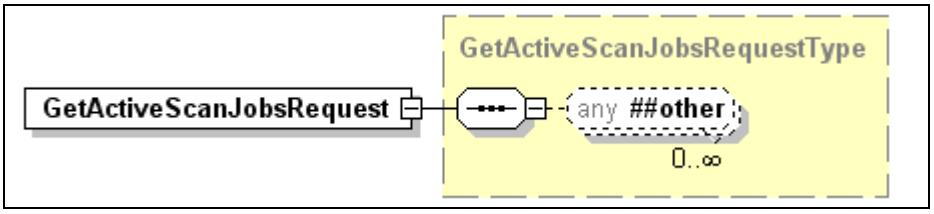
11.1.2.2 CreateScanJobResponse



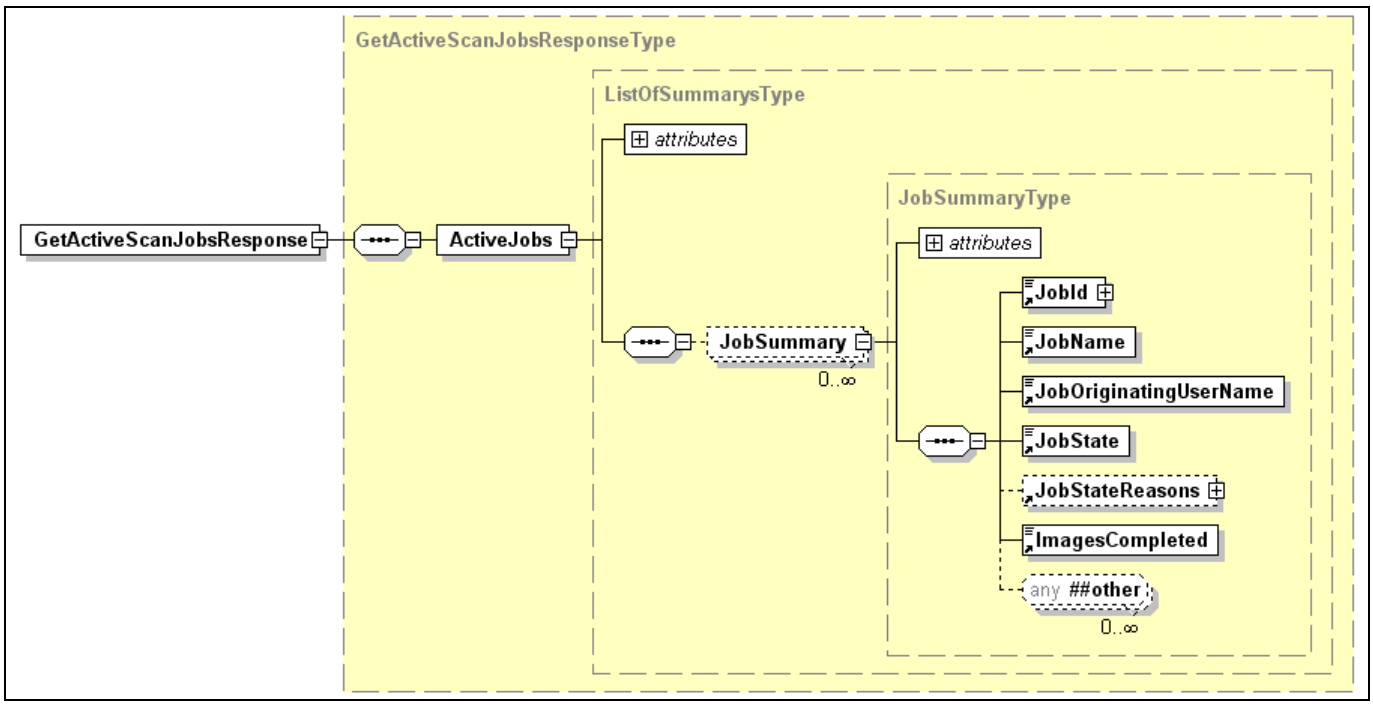
11.1.3 GetActiveScanJobs

This REQUIRED operation provides summary information on all Pending and Processing Scan Jobs.

11.1.3.1 GetActiveScanJobsRequest



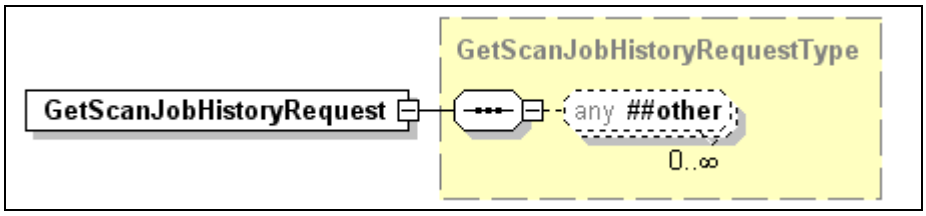
11.1.3.2 GetActiveScanJobsResponse



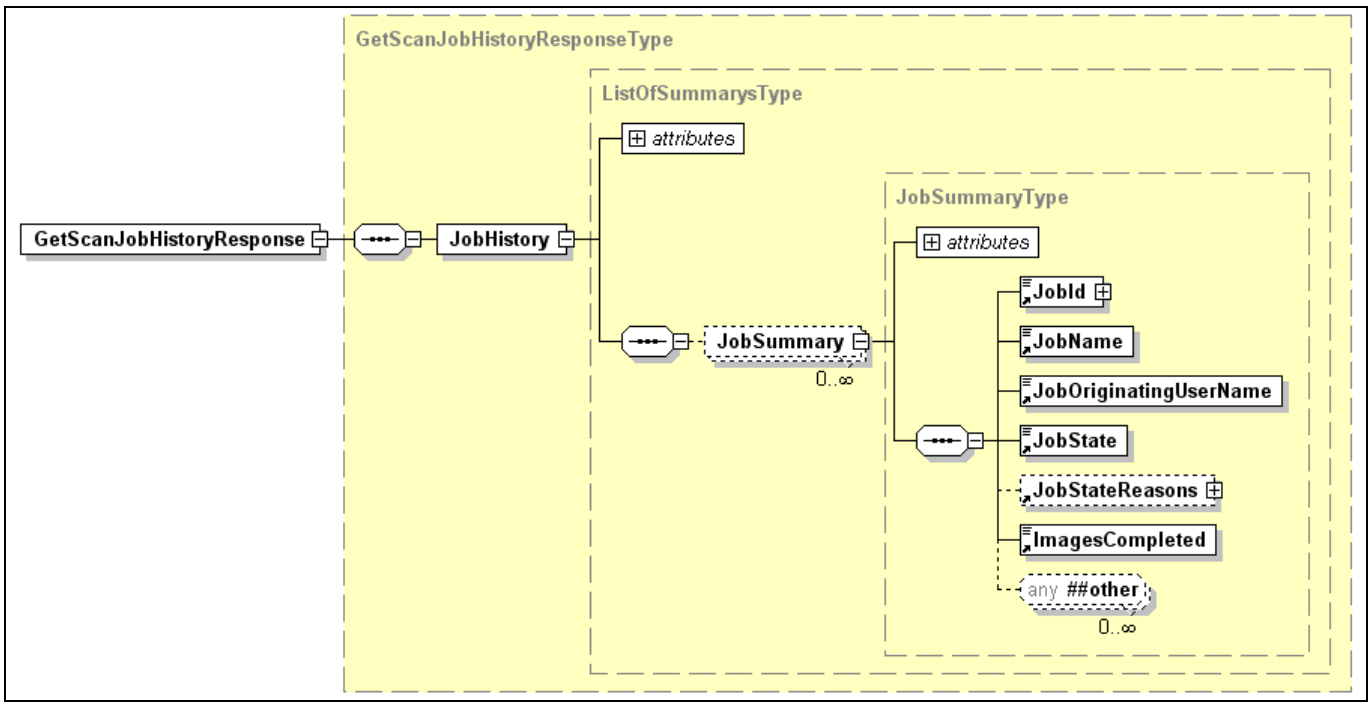
11.1.4 GetScanJobHistory

This OPTIONAL operation provides summary information on all Scan Jobs that have reached a terminating state (i.e. Completed, Canceled, Aborted).

11.1.4.1 GetScanJobHistory Request



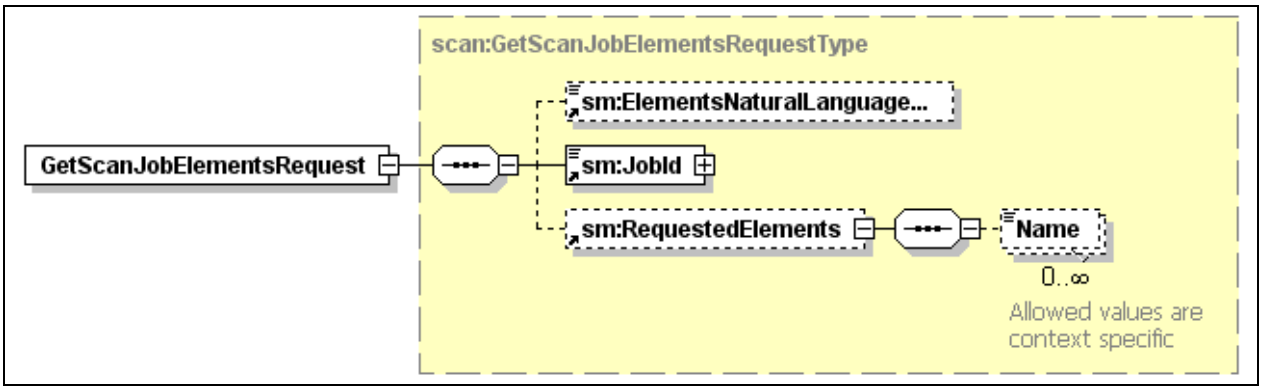
11.1.4.2 GetScanJobHistory Response



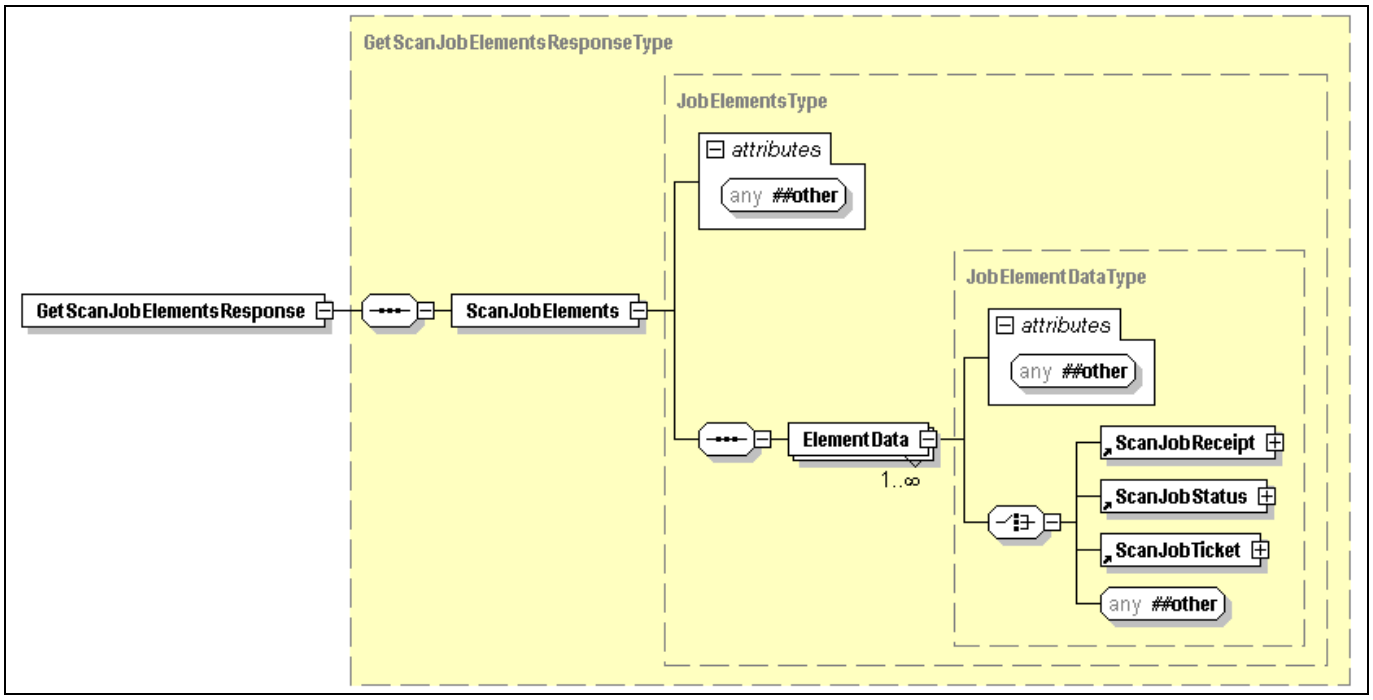
11.1.5 GetScanJobElements

This REQUIRED operation allows a Scan Client to obtain detailed information on the specified Scan Job. The Client can request specific groups of elements contained within the Scan Job. The allowed values for RequestedElements are ScanJobReceipt, ScanJobStatus, or ScanJobTicket. Vendors may extend the allowed values.

11.1.5.1 GetScanJobElements Request



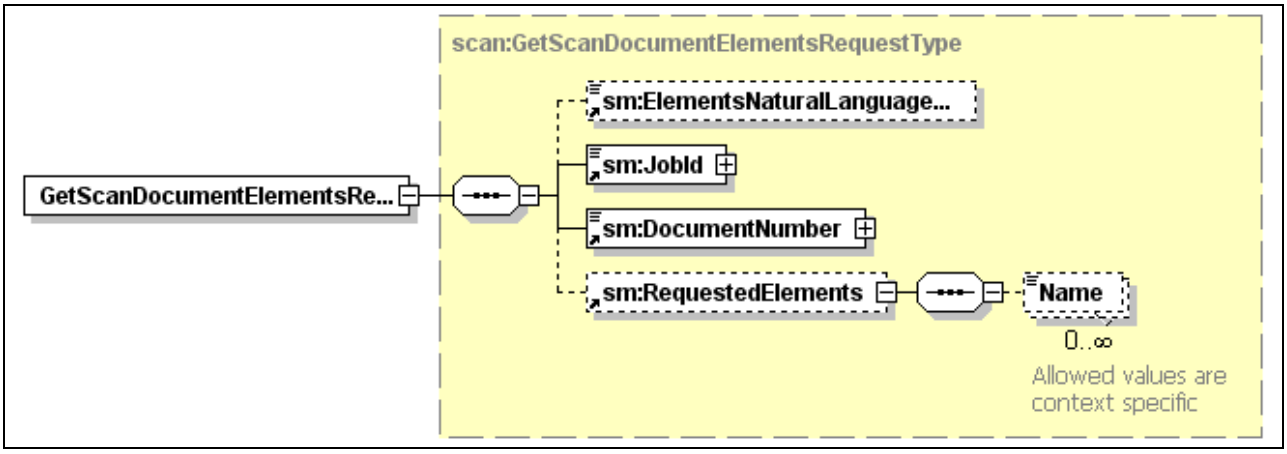
11.1.5.2 GetScanJobElements Response



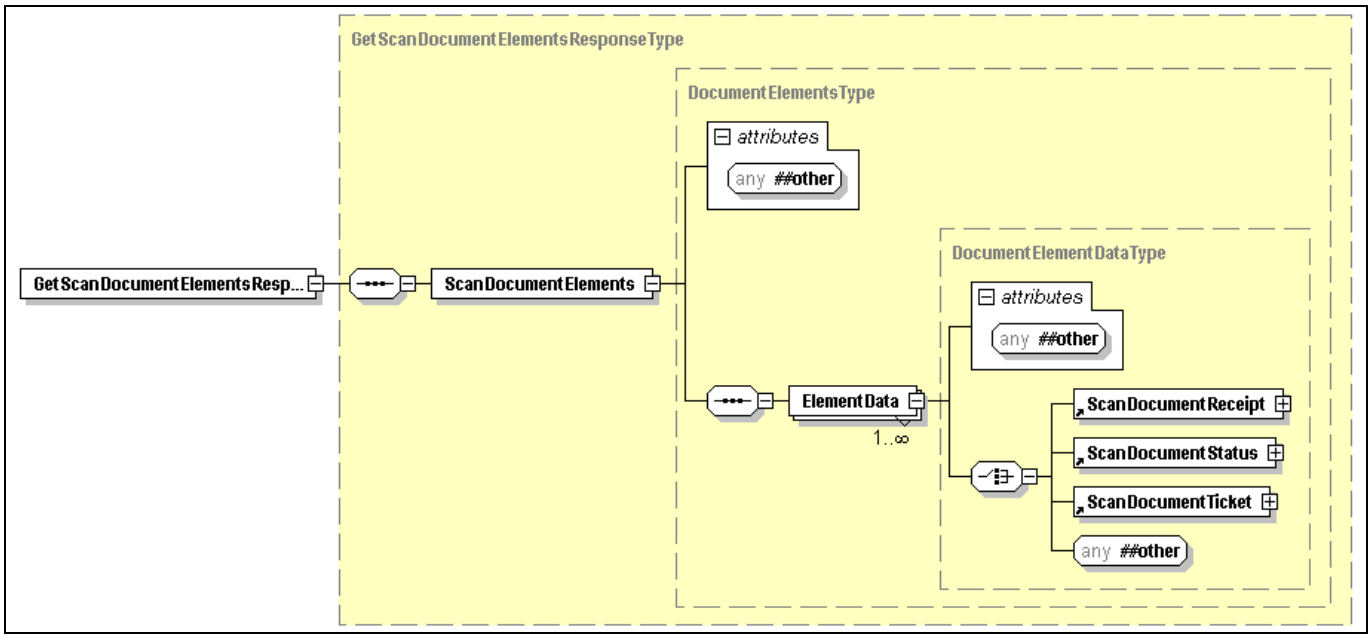
11.1.6 GetScanDocumentElements

This OPTIONAL operation allows a Scan Client to obtain detailed information on the specified Scan Document within the specified Scan Job. The Client can request specific groups of elements contained within the Scan Document. The Scan Document Data is not part of the Scan Document and can not be retrieved using this operation. However the location of the Scan Document Data is available. The allowed values for Requested Elements are ScanDocumentReceipt, ScanDocumentStatus or, ScanDocumentTicket. Vendors may extend the allowed values.

11.1.6.1 GetScanDocumentElementsRequest



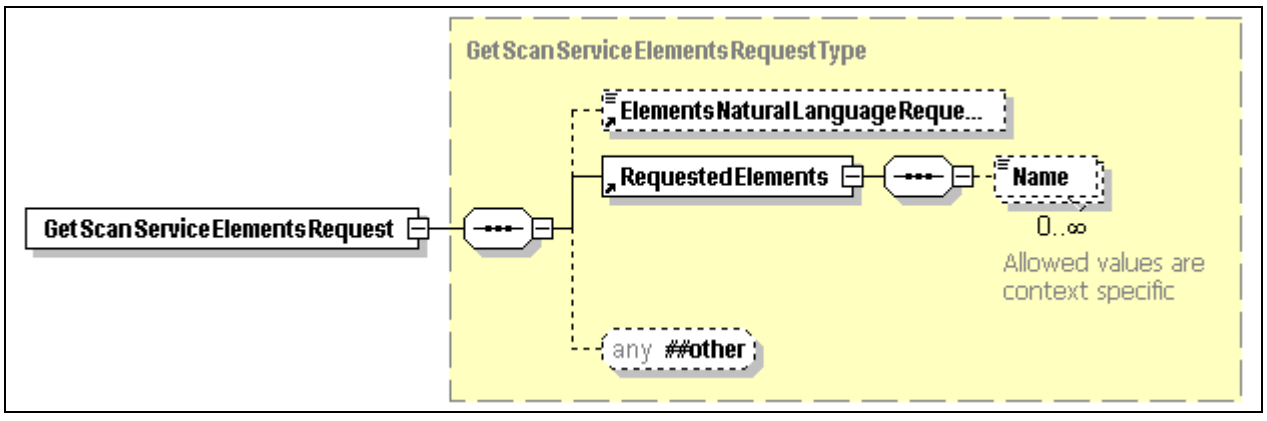
11.1.6.2 GetScanDocumentElements Response



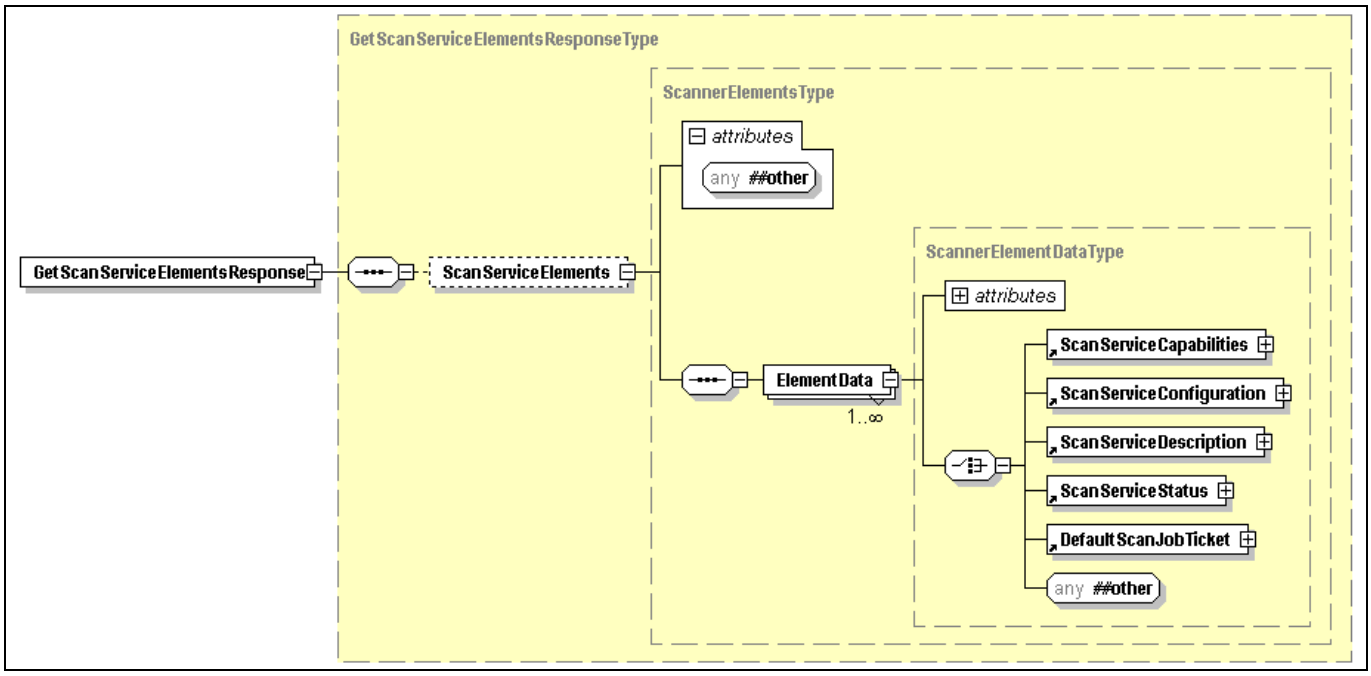
11.1.7 GetScanServiceElements

This REQUIRED operation allows a Scan Client to obtain detailed information on the Scan Service. The Client can request specific group of elements contained within the Scan Service. The allowed values for Requested Elements are ScanServiceCapabilities, ScanServiceConfiguration, ScanServiceDescription, ScanServiceStatus or DefaultScanJobTicket. Vendors may extend the allowed values.

11.1.7.1 GetScanServiceElementsRequest



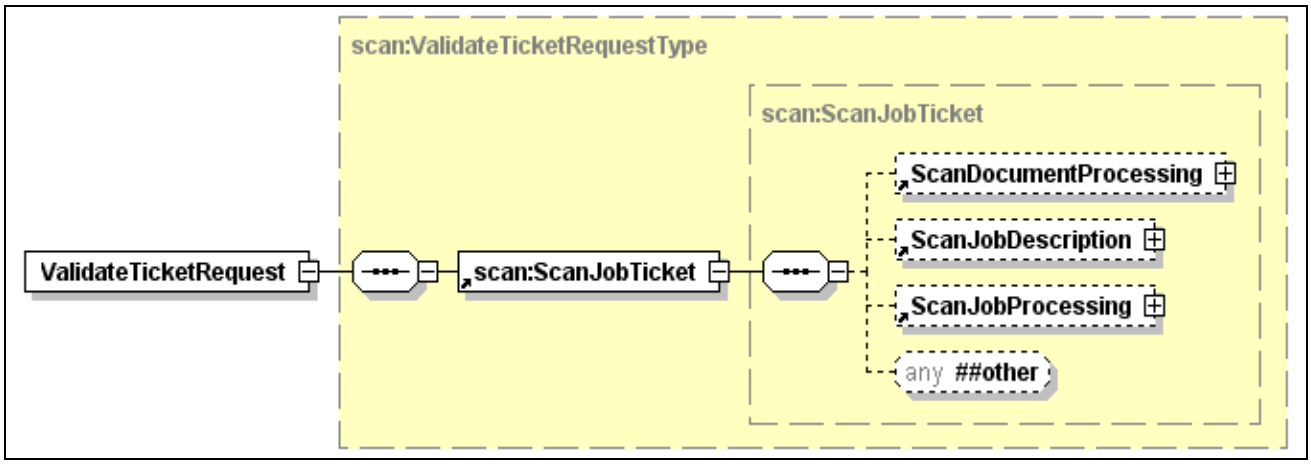
11.1.7.2 GetScanServiceElementsResponse



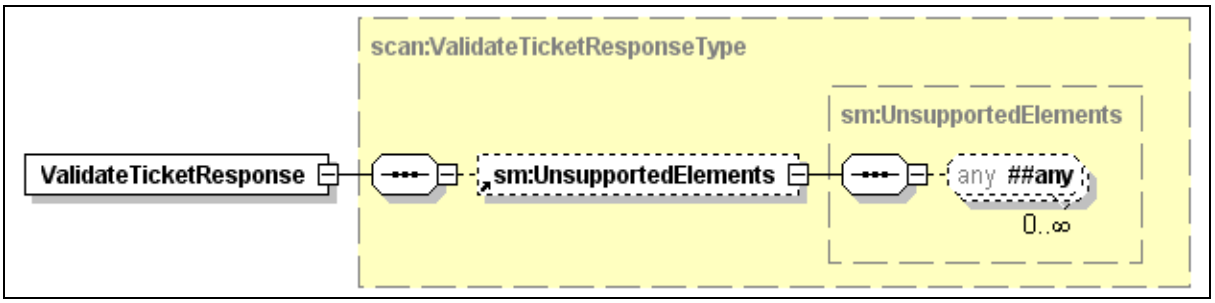
11.1.8 ValidateScanTicket

This REQUIRED operation is used to verify capabilities of a Scan Service against whatever ScanJobTicket elements are supplied by the client in the ValidateTicket request. By using the ValidateScanTicket operation a client can validate that a CreateScanJob with a ScanJobTicket would be accepted.
The response indicates if there are any unsupported elements or values. An element returned without the supplied value indicates the element is not supported by the Scan Service. An element returned with the supplied value indicates that the specific value is not supported by the Scan Service. A fault is returned for requests that are not well formed or valid.

11.1.8.1 ValidateScanTicketRequest



11.1.8.2 ValidateScanTicketResponse



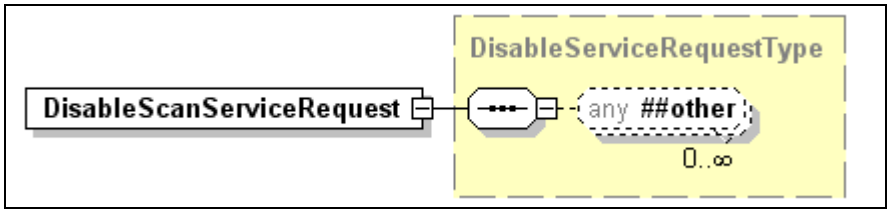
11.2 Administrative Scan Service Operations

11.2.1 DisableScanService

This operation prevents the Scan Service from accepting any new Scan Jobs (i.e. CreateScanJob operation). The Scan Service is still able to process other operations.

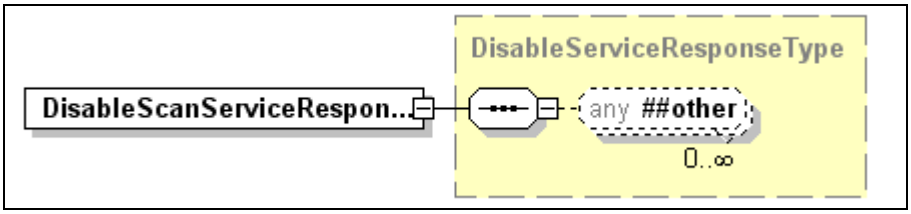
11.2.1.1 DisableScanServiceRequest

Note: The request is not expected to contain any parameters. The request below indicates that vendors may extend the request with additional information



11.2.1.2 DisableScanServiceResponse

Note: The response is expected to be an empty successful return or a fault. The response below indicates that vendors may extend the response with additional information

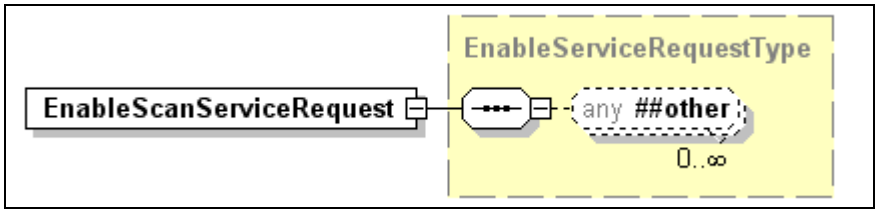


11.2.2 EnableScanService

This operation allows a previously disabled Scan Service to accepting new Scan Jobs (i.e. CreateScanJob operation).

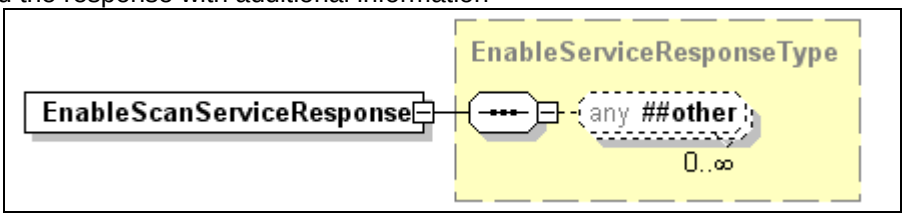
11.2.2.1 EnableScanServiceRequest

Note: The request is not expected to contain any parameters. The request below indicates that vendors may extend the request with additional information



11.2.2.2 EnableScanServiceResponse

Note: The response is expected to be an empty successful return or a fault. The response below indicates that vendors may extend the response with additional information



11.2.3 HoldScanJob

This OPTIONAL operation allows a client to hold a pending job in the queue so that it is not eligible for scheduling. If the HoldScanJob operation is supported, then the ReleaseJob operation MUST be supported, and vice-versa. The OPTIONAL "JobHoldUntil" or "JobHoldUntilTime" parameter allows a client to specify whether to hold the job until a specified time, indefinitely or until a specified time period, if supported. The Scan Service MUST accept or reject the request based on the job's current state and transition the job to the indicated new state as follows:

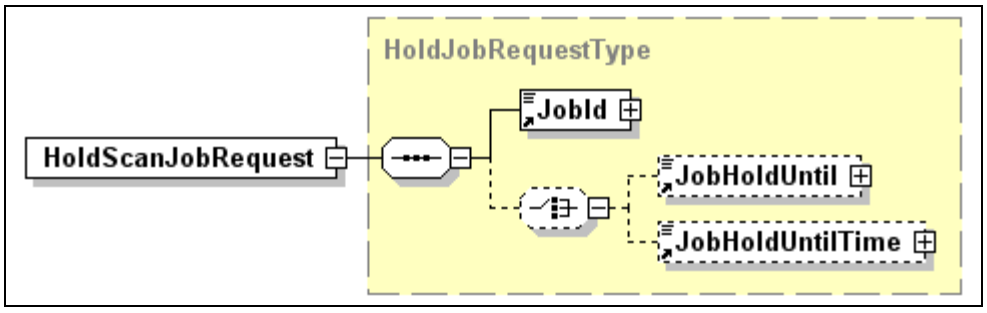
Current JobState	New JobState	Status	
Pending	PendingHeld	Success	See Note 1
Pending	Pending	Success	See Note 2
PendingHeld	PendingHeld	Success	See Note 1
PendingHeld	Pending	Success	See Note 2
Processing	Processing	Fault	
ProcessingStopped	ProcessingStopped	Fault	
Completed	Completed	Fault	
Canceled	Canceled	Fault	
Aborted	Aborted	Fault	

Note 1: If the implementation supports multiple reasons for a job to be in the PendingHeld state, the Scan Server MUST add the 'JobHoldUntilSpecified' value to the Job's "JobStateReasons" element.

Note 2: If the Scan Service supports the "JobHoldUntil" and/or the "JobHoldUntilTime" elements, but the specified time period has already started (or is the 'NoHold' value) and there are no other reasons to hold the job, the Scan Service MUST make the job be a candidate for processing immediately by putting the job in the 'Pending' state.

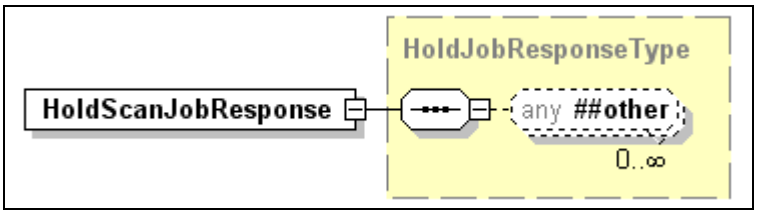
In order to keep the HoldScanJob operation simple, a request is rejected when the job is in the 'Processing' or 'ProcessingStopped' states.

11.2.3.1 HoldScanJobRequest



11.2.3.2 HoldScanJobResponse

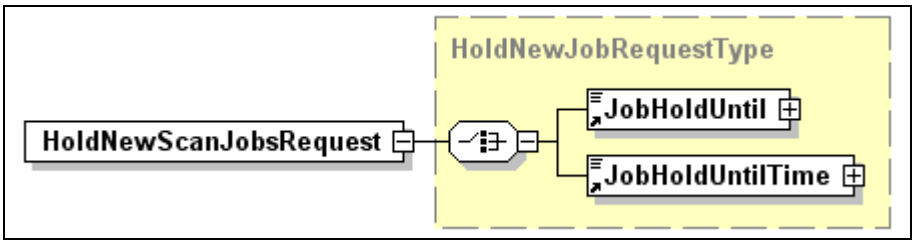
Note: The response is expected to be an empty successful return or a fault. The response below indicates that vendors may extend the response with additional information



11.2.4 HoldScanNewJobs

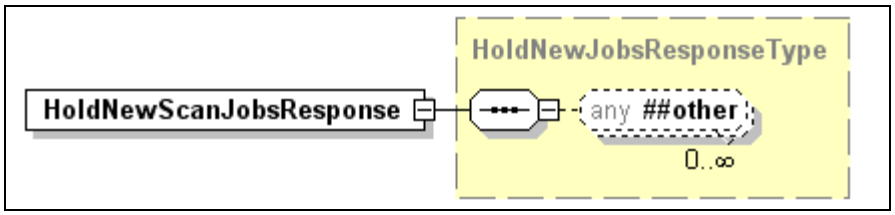
This OPTIONAL operation allows a client to prevent any new jobs from being eligible for scheduling. If the HoldNewScanJobs operation is supported, then the ReleaseNewScanJobs operation MUST be supported, and vice-versa. The OPTIONAL "JobHoldUntil" parameter allows a client to specify whether to hold new jobs until a specified time, indefinitely or until a specified time period, if supported. While this operation is in effect all new jobs MUST have the “JobHoldUntil” or the “JobHoldUntilTime” Job Processing element added with the value from the operation. The Scan Server MUST add the 'JobHoldUntilSpecified' value to the Job's "JobStateReasons” element

11.2.4.1 HoldNewScanJobsRequest



11.2.4.2 HoldNewScanJobsResponse

Note: The response is expected to be an empty successful return or a fault. The response below indicates that vendors may extend the response with additional information



11.2.5 PauseScanService

This OPTIONAL operation allows a client to stop the Scan Service from scheduling jobs. Depending on implementation, the PauseScanService operation MAY also stop the Scan Service from processing the current job. Any job that is currently being scanned is either stopped as soon as the implementation permits or is completed, depending on implementation. The Scan Service MUST still accept CreateScanJob operations to create new jobs, but MUST prevent any jobs from entering the 'Processing' state. If the PauseScanService operation is supported, then the ResumeScanService operation MUST be supported, and vice-versa.

The Scan Service stops the current job that is in the 'Processing' or 'ProcessingStopped' states as soon as the implementation permits. If the implementation will take appreciable time to stop, the Scan Service adds the ' MovingToPaused ' value to the Scan Service’s "StateReasons" element. When the Scan Service transitions to the 'Stopped' state, it removes the 'MovingToPaused' value, if present, and adds the 'Paused' value to the Scan Service’s "StateReasons" elemet. If the implementation permits the current job to stop in mid processing, the Scan Service transitions the Scan Job to the 'ProcessingStopped' state and adds the 'Stopped' value to the job's "JobStateReasons" element.

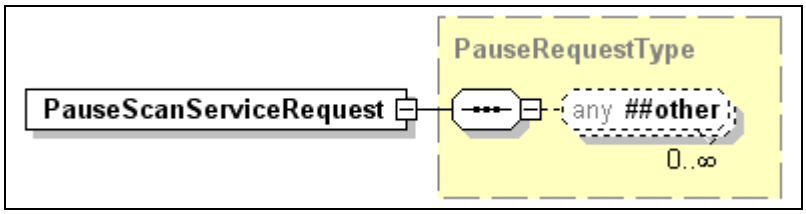
For any jobs that are 'pending' or 'pending-held', the ‘Stopped' value of the jobs' "JobStateReasons" element also applies. However, the Scan Service NEED NOT update those jobs' "JobStateReasons" element and only need return the 'Stopped' value when those jobs are queried (so-called "lazy evaluation").

The Scan Service MUST accept the request in any state and transition the Scan Service to the indicated new "State" before returning as follows:

Current State	New State	StateReason	Status	
Idle	Stopped	Paused	Success	
Processing	Processing	MovingToPaused	Success	See Note 1
Processing	Stopped	Paused	Success	See Note 2
Stopped	Stopped	Paused	Success	

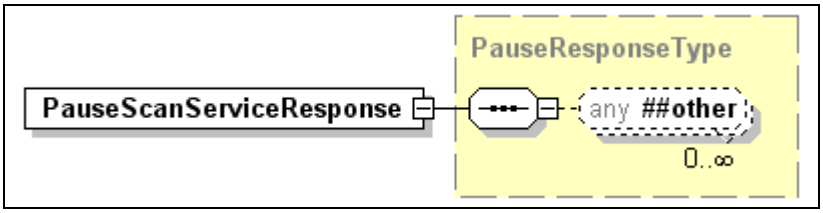
Note 1: Implementation that do not pause the current job replace the “State” ‘Processing’ value with ‘Stopped’ and replace the “StateReasons” value of ‘MovingToPaused’ with ‘Paused’.
Note 2: Implementations that are able to pause the current job.

11.2.5.1 PauseScanServiceRequest



11.2.5.2 PauseScanServiceResponse

Note: The response is expected to be an empty successful return or a fault. The response below indicates that vendors may extend the response with additional information



11.2.6 PauseScanServiceAfterCurrentJob

This OPTIONAL operation allows a client to stop the Scan Service from scheduling jobs. This operation has no effect on the current job and the Scan Service MUST complete the processing of the current job. The Scan Service MUST still accept CreateScanJob operations to create new jobs, but MUST prevent any jobs from entering the 'Processing' state. If the PauseScanServiceAfterCurrentJob operation is supported, then the ResumeScanService operation MUST be supported.

The Scan Service adds the ' MovingToPaused ' value to the Scan Service’s "StateReasons" element. When the Scan Service transitions to the 'Stopped' state, it removes the 'MovingToPaused' value, if present, and adds the 'Paused' value to the Scan Service’s "StateReasons" elemet.

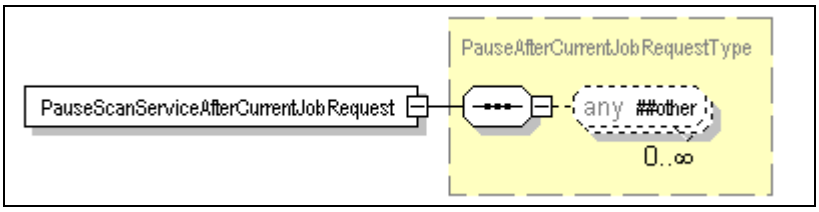
For any jobs that are 'pending' or 'pending-held', the ‘Stopped' value of the jobs' "JobStateReasons" element also applies. However, the Scan Service NEED NOT update those jobs' "JobStateReasons" element and only need return the 'Stopped' value when those jobs are queried (so-called "lazy evaluation").

The Scan Service MUST accept the request in any state and transition the Scan Service to the indicated new "State" before returning as follows:

Current State	New State	StateReason	Status	
Idle	Stopped	Paused	Success	
Processing	Processing	MovingToPaused	Success	See Note 1
Stopped	Stopped	Paused	Success	

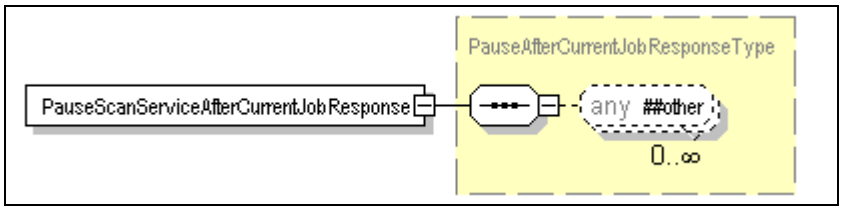
Note 1: Once the currently processing job completes the State will transition to Stopped and the MovingToPaused StateReason will be remove and replaced with Paused

11.2.6.1 PauseScanServiceAfterCurrentJobRequest



11.2.6.2 PauseScanServiceAfterCurrentJobResponse

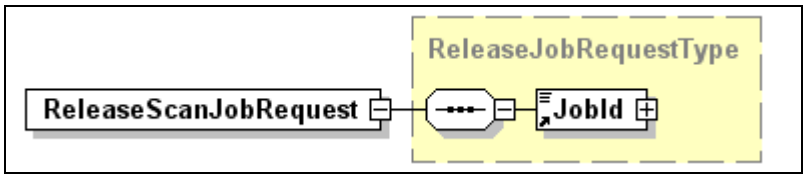
Note: The response is expected to be an empty successful return or a fault. The response below indicates that vendors may extend the response with additional information



11.2.7 ReleaseScanJob

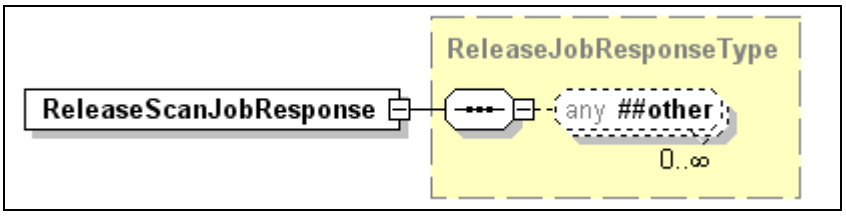
This OPTIONAL operation allows a client to release a previously held job the queue so that it is eligible for scheduling. If the ReleaseJob operation is supported, then the HoldScanJob operation MUST be supported, and vice-versa.

11.2.7.1 ReleaseScanJobRequest



11.2.7.2 ReleaseScanJobResponse

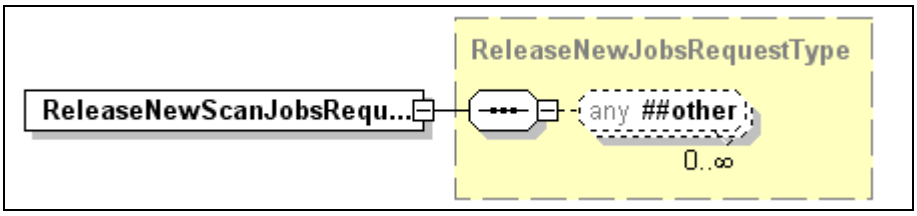
Note: The response is expected to be an empty successful return or a fault. The response below indicates that vendors may extend the response with additional information



11.2.8 ReleaseNewScanJobs

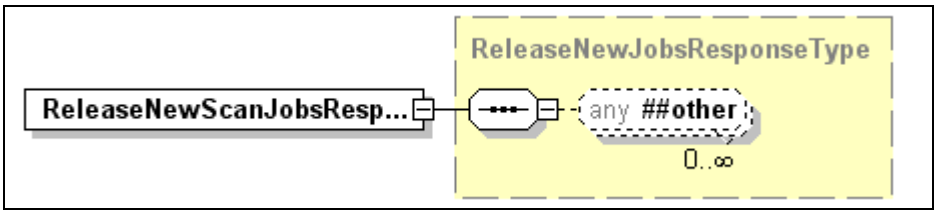
This OPTIONAL operation allows a client to release previously held new jobs so they are now eligible for scheduling. If the ReleaseNewScanJobs operation is supported, then the HoldNewScanJobs operation MUST be supported, and vice-versa. The Scan Server MUST remove the 'JobHoldUntilSpecified' value from the Job's "JobStateReasons" element

11.2.8.1 ReleaseNewScanJobsRequest



11.2.8.2 ReleaseNewScanJobsResponse

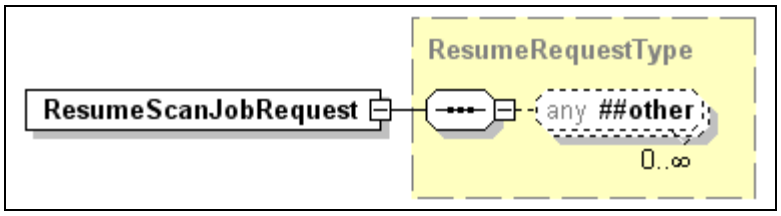
Note: The response is expected to be an empty successful return or a fault. The response below indicates that vendors may extend the response with additional information



11.2.9 ResumeScanService

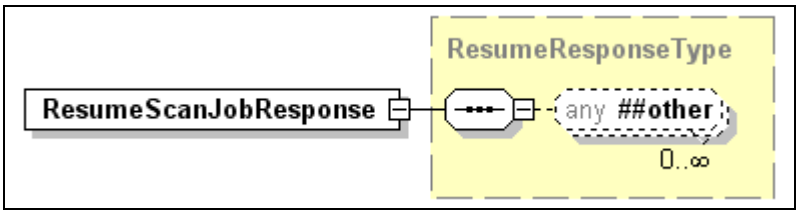
This OPTIONAL operation allows a client to cause the Scan Service to begin scheduling jobs after scheduling has been paused. If the ResumeScanService operation is supported, then the PauseScanService operation MUST be supported, and vice-versa. When succewssfully processing this operation the Scan Service removes the 'Paused' value to the Scan Service's "StateReasons" elemet.

11.2.9.1 ResumeScanServiceRequest



11.2.9.2 ResumeScanServiceResponse

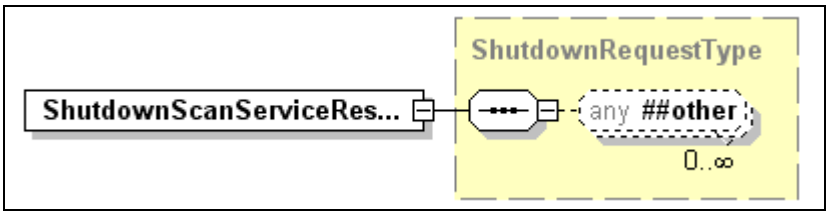
Note: The response is expected to be an empty successful return or a fault. The response below indicates that vendors may extend the response with additional information



11.2.10 ShutdownScanService

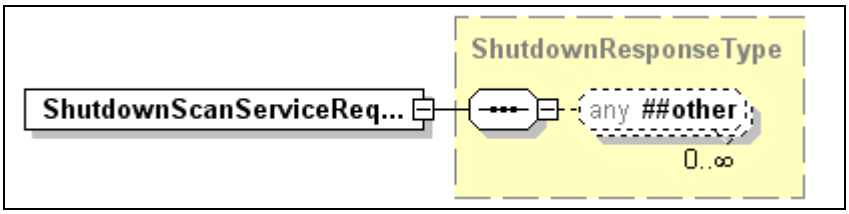
This OPTIONAL administrative operation causes the Scan Service to stop scheduling jobs, complete the processing of any currently processing jobs as soon as the implementation allows, stop accepting new jobs, and causes the Scan Service to terminate in an orderly fashion.

11.2.10.1 ShutdownScanServiceRequest



11.2.10.2 ShutdownScanServiceResponse

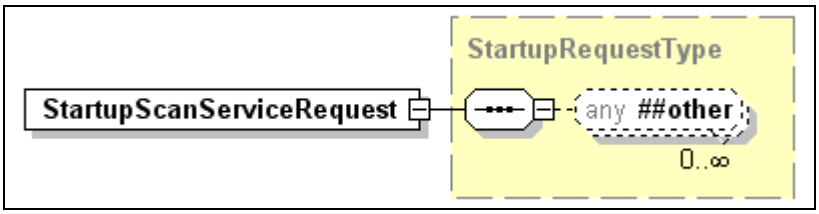
Note: The response is expected to be an empty successful return or a fault. The response below indicates that vendors may extend the response with additional information



11.2.11 StartupScanService

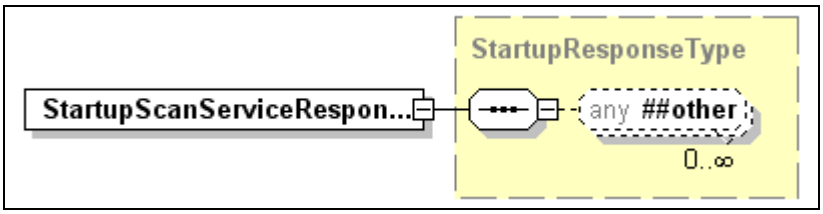
This OPTIONAL administrative operation causes a new instance of a Scan Service to to begin initialization and then move to a processing state if possible (i.e. no hardware or software errors occur that would prevent normal operation).

11.2.11.1 StartupScanServiceRequest



11.2.11.2 StartupScanServiceResponse

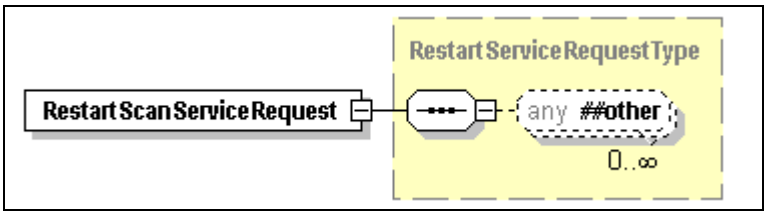
Note: The response is expected to be an empty successful return or a fault. The response below indicates that vendors may extend the response with additional information



11.2.12 RestartScanService

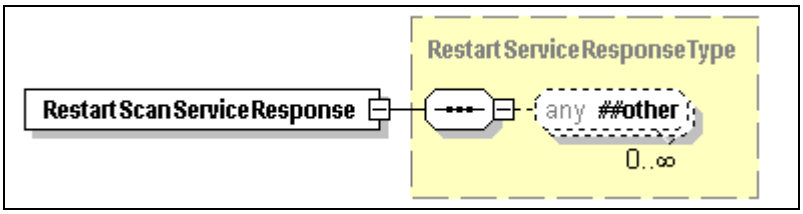
This OPTIONAL administrative operation causes a previously shut down instance of a Scan Service to to begin initialization and then move to a processing state if possible (i.e. no hardware or software errors occur that would prevent normal operation).

11.2.12.1 RestartScanServiceRequest



11.2.12.2 RestartScanServiceResponse

Note: The response is expected to be an empty successful return or a fault. The response below indicates that vendors may extend the response with additional information



12 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.

12.1 Client Conformance Requirements

A conforming client MUST support all REQUIRED operations as defined in this document. For each parameter included in an operation request, a conforming client MUST supply a value whose type and value syntax conforms to the requirements of the Model document as specified in Sections 11. A conforming client MAY supply any extensions in an operation request, as long as they meet the requirements in Section 12.4.

When sending a request, a conforming client NEED NOT supply any parameters that are indicated as OPTIONALLY supplied by the client.

A 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 Scan Service

An operation response may contain elements and/or values that the client does not expect. Therefore, a client implementation MUST gracefully handle such responses and not refuse to inter-operate with a conforming Scan Service that is returning extended elements and/or values that conform to Section 12.4. Clients may choose to ignore any parameters, elements, or values that they do not understand.

12.2 Scan Service Conformance Requirements

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

12.2.1 Objects

Conforming implementations MUST implement all of the model objects as defined in this specification in the indicated sections:

Section 7- Scan Service
Section 8- Scan Job

12.2.2 Operations

Conforming Scan Service implementations MUST implement all of the REQUIRED model operations, including REQUIRED responses, as defined in this specification in the indicated sections:

CancelScanJob (section 11.1.1) REQUIRED
CreateScanJob (section 11.1.2) REQUIRED
GetActiveScanJobs (section 11.1.3) REQUIRED
GetScanJobElements (section 11.1.5) REQUIRED
GetScanJobHistory (section 11.1.4) REQUIRED
GetScanServiceElements (section 11.1.7) REQUIRED
ValidateScanTicket (section 11.1.8) REQUIRED

Conforming Scan Service MUST support all REQUIRED operation elements and all values of such elements if so indicated in the description. Conforming Scan Service MUST ignore all unsupported or unknown operation elements received in a request, but MUST reject a CreateScanJob request that contains an unknown element that contains the MustHonor attribute with a value of 'true'.

12.3 Scan Service Elements

Conforming Scan Service 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 12.4. 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.

12.4 Extensions

Conforming Scan Service 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.

13 PWG Registration Considerations

Once the specification and associated schema is published it will require a new version of the specification to register extensions to the ScanService model. Vendors may use extensions in their own namespace until such time as an update to the specification is under review. At that time the extension can be registered with the PWG and included in the PWG specification.

14 Internalization Considerations

All element values defined by enumeration (e.g. State) represent keywords. Keywords are never localized by the device. The client may convert the values into a form acceptable to the client. This includes not only localization but also transformations into graphical representation. The elements with an extensible list of keyword are represented by the union of an enumeration of keywords and a pattern for new values.

Some of the elements have values that are Scan Service generated strings (e.g. StateMessages). In each operation request, the client identifies a natural language that affects the Scan Service generated strings returned by the Scan Service in operation responses. The Scan Service MUST provide the localized value as requested by the user for any supported natural languages. A request for a language not supported results in a response with the string in the default localization.
The final category of string values are those supplied by administrator or End User (e.g. JobName). No localization is performed on these strings and they are returned in operation responses as set by the administrator or End User.

15 Security Considerations

15.1 Storing Scan Documents in a Document Repository

Organizations with higher security requirements may require End Users to store their Documents only in the designated Document Repositories for which organizational document access control policies can easily be instrumented. It is the End User's responsibility to ensure that their target document repositories has been configured to support the Scan Service writing user's Scan Document Data into the repository. This implies the requesting user has been authenticated in the same network domain of the Document Repository.

15.2 Protection of End User's Scan Documents

An End User's Scan Documents can be protected from disclosure by encrypting the content of the Documents and protected from modification by signing the content of the Documents when these Documents are stored in a repository or being transmitted over a communication link.

Signing or encrypting Documents stored in a Document 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 Documents stored in a Document Repository is outside the scope of the Scan Service. It is RECOMMENDED that the End User designates a Document Repository that has their desired level of signing or encryption capabilities.

15.3 Restricted Use of Scan Service Features

The management of the site policy for the use of Scan Service features is outside the scope of this specification.

16 References

16.1 Normative References

[IANA_NUMBERS]
IANA Assigned Numbers, On-line database replaces RFC3232, current revision October 9, 2008, http://www.iana.org/assignments/ianaiftype-mib

[ISO 8601:2000 Second Edition]
ISO (International Organization for Standardization). Representations of dates and times, second edition, December 15, 2000, K. Ocke, T. Hastings, ftp://ftp.pwg.org/pub/pwg/candidates/cs-ippprodprint10-20010212-5100.3.pdf

[PWG5100.3]
PWG 5100.3-2001, "Internet Printing Protocol (IPP): Production Printing Attributes - Set1", February 12, 2001, K. Ocke, T. Hastings, ftp://ftp.pwg.org/pub/pwg/candidates/cs-ippprodprint10-20010212-5100.3.pdf

[PWG5100.5]
PWG 5100.5-2003, "Internet Printing Protocol (IPP): Document Object", October 31, 2003, D. Carney, T. Hastings, and P. Zehler, ftp://ftp.pwg.org/pub/pwg/candidates/cs-ippdocobject10-20031031-5100.5.pdf

[PWG5101.1]
PWG 5101.1-2002 Media Standardized Names, February 26, 2002
ftp://ftp.pwg.org/pub/pwg/candidates/cs-pwgmsn10-20020226-5101.1.pdf

[PWG5106.1-2007]
PWG 5106.1-2007, "The Printer Working Group (PWG) Standardized Imaging System Counters 1.1", April 27, 2007, H. Lewis, I. McDonald, J. Thrasher, W. Wagner, and P. Zehler, ftp://ftp.pwg.org/pub/pwg/candidates/cs-wimscount11-20070427-5106.1.pdf

[RFC2119]
S. Bradner, “Key words for use in RFCs to Indicate Requirement Levels”, RFC 2119, March 1997. http://www.ietf.org/rfc/rfc2119.txt

[RFC 2426]
F. Dawson, T. Howes, “vCard MIME Directory Profile”, RFC 2426, September 1998. http://www.ietf.org/rfc/rfc2426.txt

[RFC 2782]
A. Gulbrandsen, P. Vixie, L. Esibov, ” A DNS RR for specifying the location of services (DNS SRV)”, RFC 2782, February 2000. http://www.ietf.org/rfc/rfc2782.txt

[RFC 2790]
S. Waldbusser, P. Grillo, “Host Resources MIB”, RFC 2426, March 2000.

[RFC 2863]
K. McCloghrie, F. Kastenholz, “The Interfaces Group MIB”, RFC 2863, June 2000. http://www.ietf.org/rfc/rfc2863.txt

[rfc2911]
RFC 2911 “Internet Printing Protocol/1.1 Model and Semantics”, September 2000, T. Hastings, R. Herriot, R. deBry, S. Isaacson, P. Powell, ftp://ftp.rfc-editor.org/in-notes/rfc2911.txt

[rfc3066]
RFC 3066 “Tags for the Identification of Languages”, January 2001, H. Alvestrand, http://www.ietf.org/rfc/rfc3066.txt

[rfc3629]
RFC 3629 “UTF-8, a transformation format of ISO 10646”, November 2003, T. F. Yergeau, http://www.rfc-editor.org/rfc/rfc3629.txt

[RFC 3712]
P. Fleming, I. McDonald, “Lightweight Directory Access Protocol (LDAP): Schema for Printer Services”, RFC 3712, February 2004. http://www.ietf.org/rfc/rfc3712.txt

[RFC 3805]
R. Bergman, H. Lewis, I. McDonald, “Printer MIB v2”, RFC 3805, June 2004. http://www.ietf.org/rfc/rfc3805.txt

[RFC 3806]
R. Bergman, H. Lewis, I. McDonald, “Printer Finishing MIB”, RFC 3806, June 2004. http://www.ietf.org/rfc/rfc3806.txt

[RFC 3966]
H. Schulzrinne, “The tel URI for Telephone Numbers”, RFC 3966, December 2004. http://www.ietf.org/rfc/rfc3966.txt

[rfc3986]
RFC 3986 “Uniform Resource Identifier (URI): Generic Syntax”, January 2005, T. Berners-Lee, R. Fielding, L. Masinter, http://www.rfc-editor.org/rfc/rfc3986.txt

[RFC 4355]
R. Brandner, L. Conroy, R. Stastny , “IANA Registration for Enumservices email, fax, mms, ems, and sms”, RFC 4355, January 2006. http://www.ietf.org/rfc/rfc4355.txt

[rfc4395]
RFC 4395 “Guidelines and Registration Procedures for New URI Schemes”, February 2006, T. Hansen, T. Hardie, L. Masinter, http://www.rfc-editor.org/rfc/rfc4395.txt

[UNICODE]
“The Unicode Standard 5.0”, October 2006, The Unicode Consortium, ISBN 0-321-48091-0 http://www.unicode.org/versions/Unicode5.1.0/

[XML]
“Extensible Markup Language (XML) 1.0 (Fifth Edition)”, November 26, 2008, W3C Recommendation, T. Bray, A. Malhotra, http://www.w3.org/TR/REC-xml/

[WS-Scan]
“Scan Service DefinitionVersion 1.0 For Web Services on Devices”, November 2006, Microsoft, M Fenelon, http://www.microsoft.com/whdc/connect/rally/wsdspecs.mspx

[XML]
“Extensible Markup Language (XML) 1.0 (Fifth Edition)”, November 26, 2008, W3C Recommendation, T. Bray, A. Malhotra, http://www.w3.org/TR/REC-xml/

[XSD]
“XML Schema Part 2: Datatypes Second Edition”, October 28, 2004, W3C Recommendation, P Biron, J. Paoli, . Sperberg-McQueen, E. Maler, F. Yergeau, http://www.w3.org/TR/xmlschema-2/

16.2 Informative References

[prod-print2]
T. Hastings, and D. Fullman, “Internet Printing Protocol (IPP): Production Printing Attributes - Set 2”, August 21, 2002, ftp://ftp.pwg.org/pub/pwg/ipp/new_PPE/pwg-ipp-prod-print-set2-draft-v0_1-020821.pdf ,

[DES]
FIPS PUB 46-3 Federal Information Processing Standards Publication “Data Encryption Standard (DES)”, October 1999, http://csrc.nist.gov/publications/fips/fips46-3/fips46-3.pdf

[AES]
FIPS PUB 197 Federal Information Processing Standards Publication 197 “Advanced Encryption Standard (AES)”, November 2001, http://csrc.nist.gov/publications/fips/fips197/fips-197.pdf

[ECC]
RFC 1321 “TLS Elliptic Curve Cipher Suites with SHA-256/384 and AES Galois Counter Mode”, May 2008, E. Rescorla, http://www.ietf.org/internet-drafts/draft-ietf-tls-ecc-new-mac-07.txt

[MD5]
RFC 1321 “The MD5 Message-Digest Algorithm”, April 1992, R. Rivest, http://www.ietf.org/rfc/rfc1321.txt

[MD4]
RFC 1320 “The MD4 Message-Digest Algorithm”, April 1992, R. Rivest, http://www.ietf.org/rfc/rfc1320.txt

[MD2]
RFC 1319 “The MD2 Message-Digest Algorithm”, April 1992, B. Kaliski, http://www.ietf.org/rfc/rfc1319.txt

[SHA]
FIPS PUB 180-1 Federal Information Processing Standards Publication 180-1 “Secure Hash Standard”, April 1995, http://www.itl.nist.gov/fipspubs/fip180-1.htm

[CHMOD]
The Open Group Base Specifications Issue 6 IEEE Std 1003.1 POSIX - chmod() function, 2004, http://www.opengroup.org/onlinepubs/000095399/functions/chmod.html

[RFC 1696]
J. Barnes, L. Brown, R. Royston, S. Waldbusser, “Modem Management Information Base (MIB) using SMIv2”, RFC 1696, August 1994. http://www.ietf.org/rfc/rfc1696.txt

[STAT]
The Open Group Base Specifications Issue 6 IEEE Std 1003.1 POSIX - stat() function, 2004, http://www.opengroup.org/onlinepubs/000095399/basedefs/sys/stat.h.html

17 Author’s Address

Nancy Chen
Solutions and Technology
Oki Data
2000 Bishops Gate Blvd.
Mt. Laurel, NJ 08003
Phone: 856 222 7006
Fax: 856 222 5130
Email: nchen@okidata.com

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:

Shah Bhatti - Samsung
Lee Farrell – Cannon
Mike Fenelon – Microsoft
Brett Green – HP
Nam Heo - Samsung
Harry Lewis – IBM
Ira McDonald – High North
Glen Petrie – Epson
Andrey Savov – Toshiba
Jerry Thrasher – Lexmark
David Whitehead - Lexmark
Bill Wagner – TIC
Craig Whittle – Sharp

18 Apendix A – WS-Scan protocol mapping

This model can be mapped onto Microsoft's WS-Scan protocol. The intent is not to map all of the PWG Model but to map those parts of the PWG model that overlap the WS-Scan model. There are a number of mapping issues that must be addressed. Below is a summary of some of those mapping issues. This list is not meant to be exhaustive but does help clarify the WS-Scan mapping.

1. The target namespace must be Microsoft's and not the target namespace in the PWG's WSDL and schema files associated with this specification.
2. The PWG model for the transfer of scan data is a push model (i.e. the Scan Service delivers the scan data to a location specified by the requesting client). Microsoft uses a pull model where the scan data is retrieved by an explicit operation (i.e. RetrieveImage).
    a. Although the PWG operation model does not include this operation it is possible to provide this extended operation.
    b. The information required for the retrieving the Image (i.e. DestinationToken) can be held in an extension to the PWG Destination element.
    c. The PWG Model does not specify scan events. WS-Scan requires an event to coordinate the timely delivery of scan data to the WS-Scan client. This can be handled through protocol binding specific extensions. The PWG model does represent the state transition required to trigger the ScanAvailableEvent
3. There are a number of PWG elements that have the same semantics and syntax as associated WS-Scan elements. They differ only in the element name. Below are the PWG elements and the associated WS-Scan elements
    a. PWG "ScanService" maps to WS-Scan "Scanner"
    b. PWG "CurrentTime" maps to WS-Scan "ScannerCurrentTime"
    c. PWG "ServiceName" maps to WS-Scan "ScannerName"
    d. PWG "ServiceInfo" maps to WS-Scan "ScannerInfo"
    e. PWG "ScanService.ScanServiceCapabilities" maps to WS-Scan "Scanner.ScannerConfiguration.DeviceSettings"
        i. PWG "DocumentFormat" maps to WS-Scan "FormatSupported"
        ii. PWG "DocumentSizeAutoDetect" maps to WS-Scan "DocumentSizeAutoDetectSupported"
        iii. PWG "Brightness" maps to WS-Scan "BrightnessSupported"
        iv. PWG "Contrast" maps to WS-Scan "ContrastSupported"
        v. PWG "Rotation" maps to WS-Scan "RotationSupported"

vi. PWG “CompressionQualityFactor” maps to WS-Scan “CompressionQualityFactorSupported”

f. PWG “ImagesCompleted” maps to WS-Scan “ScansCompleted”
g. PWG “DateTimeAtCreation” maps to WS-Scan “JobCreatedTime”
h. PWG “DateTimeAtCompleted” maps to WS-Scan “JobCompletedTime”
i. PWG “ScanService.ScanServiceCapabilities.ScanDocumentProcessing.sides” maps to WS-Scan “Scanner.ScannerConfiguration.ADF.SupportsDuplex”
j. PWG “ScanService.DefaultScanJobTicket.ScanDocumentProcessing” maps to WS-Scan “Scanner.DefaultScanTicket.DocumentParameters”
k. PWG “ScanJob.ScanJobReceipt” maps to WS-Scan “Job.Document.DocumentFinalParameters”

4. There are some items whose syntax differ but the semantics are directly mapped
   a. PWG “ScanService.ScanServiceConfiguration.ScanMediaPaths” (“ScanMediaPathType”) maps to WS-Scan “Scanner.ScannerConfiguration.[Platen, ADF, Film]”
      i. PWG “ScanMediaPathMaxMediaSizeName” and “ScanMediaPathMaxMediaSizeName” that use self describing media names maps to WS-Scan “Min” and “Max” sizes for Platen, ADF and Film that use “Height” and Width”
   b. PWG ScanService.ScanServiceConfiguration.Scanner.ScannerStatus.Addressabilties” maps to WS-Scan OpticalResolution for ADF, Platen and Film

